

Affine Invariant Image Comparison and Its Applications

Vadim Fedorov

TESI DOCTORAL UPF / 2016

Director de la tesi
Prof. Dra. Coloma Ballester,
Department of Information and Communication Technologies



To Tati

—

—

|

|

|

Acknowledgements

I wish I could thank Vicent Caselles for the opportunity he gave me. Our acquaintance was sadly brief, but without any doubts priceless...

First of all I would like to thank my thesis supervisor Coloma Ballester for her constant support, trust and guidance during these tough but exciting four years.

Many thanks to my collaborators Pablo Arias, Gabriele Facciolo and Rida Sadek, who contributed a lot to this work with their knowledge and experience.

I am thankful to my colleagues Maria Oliver, Roberto Palomares, Juan Francisco Garamendi, Olga Slizovskaya, Gloria Haro, Felipe Calderero, Juan Calvo and Vanel Lazcano for the friendly and cozy atmosphere in the group. I had a great time traveling together with Maria and Roberto. And coffee breaks with Olga were very helpful.

Of course, I would also like to thank my family and my close friends from Russia for their support and encouragement.

And finally, special thanks to Tati, who has made this whole adventure possible.

Abstract

English

Image comparison is a main ingredient in many image processing and computer vision problems and applications, and not surprisingly it is a very diverse topic. The subject of this thesis is the comparison of local *patches* of images by means of similarity measures (or distance functions). In particular, we are interested in *affine invariant* patch-wise image comparison which opens the door to a more thorough analysis of similarities and self-similarities present in natural images. Our work is based on a recently proposed axiomatic framework for similarity measures between images defined on Riemannian manifolds. At the beginning we derive and study some affine invariant similarity measures and then present two novel methods built around them. The first method for exemplar-based image inpainting is aimed at the recovery of occluded, missing or corrupted parts of an image, in such a way that the reconstructed image looks natural. It is capable of reconstructing textures under perspective or even more complex distortions. The second method extends the well-known Non-Local Means approach for image denoising by taking advantage of affine invariant self-similarities of real images. Our extension improves the original method in both quantitative and qualitative assessments, and the results are promising when compared with state-of-the-art methods.

Español

La comparación de imágenes es un ingrediente fundamental en muchos problemas de procesamiento de imagen y visión por computador. Esta tesis aborda el problema de la comparación de entornos locales en imágenes, o *patches*, por medio de medidas de similitud (o funciones distancia). En particular, estudiamos el problema de la comparación invariante afín de

imágenes a partir de sus patches, lo cual abre la puerta a un análisis más profundo de la estructura de similitud y auto-similitud existente en imágenes naturales. Nuestro trabajo parte de una aproximación axiomática reciente a las medidas de similitud entre imágenes definidas en variedades de Riemann. Empezamos obteniendo y estudiando medidas de similitud afín invariantes para después construir con ellas dos nuevos métodos. El objetivo del primero de ellos es la reconstrucción o completación plausible de regiones de una imagen donde la información se ha perdido, dañado o está oculta. El modelo propuesto es capaz de reconstruir texturas con distorsión perspectiva o incluso más compleja. El segundo método extiende la aproximación denominada de Non-Local Means para el problema de eliminación de ruido en imágenes aprovechando la auto-similitud invariante afín de las imágenes reales. Nuestra extensión es comparada con éxito con el método original, tanto cualitativa como cuantitativamente, y se obtienen resultados prometedores en comparación con los métodos del estado del arte.

Preface

In short, the subject of this thesis is the comparison of a pair of locations in an image or two images. In digital image processing it is common to see an image as a grid of pixels, each of which contains one or several numbers describing the color at that point. In a regular color image every pixel carries 24 bits of color information which in total makes $16\,777\,216$ possible colors. Even though this number might seem large, the color of a single pixel is not very distinctive to use it alone in the comparison. A logical idea would be to consider a small neighborhood of pixels around a given location. Even a tiny neighborhood 3 by 3 pixels in size already gives around 10^{65} different combinations to encode local structure and thus is much more informative. In the image and video processing literature these neighborhoods are usually called *patches*. A function that compares two given patches can be called *patch similarity measure* or *patch distance measure*. Essentially, in the context of image comparison, *similarity* and *distance* represent the same relation between two patches in the whole space of patches; therefore, we shall use these terms interchangeably.

Image comparison by means of patch similarity measures has multitude of fascinating applications. Obviously, different images of the same scene contain resembling visual details. Moreover, natural images themselves are normally redundant and self-similar. Even though patches capture only small local pieces of visual information, by comparing them we can reveal global geometric structures and texture patterns. In other words, patch-wise image comparison allows us to analyze similarities and self-similarities in natural images.

Frequently, patches that are used in comparison are rigid in the sense that

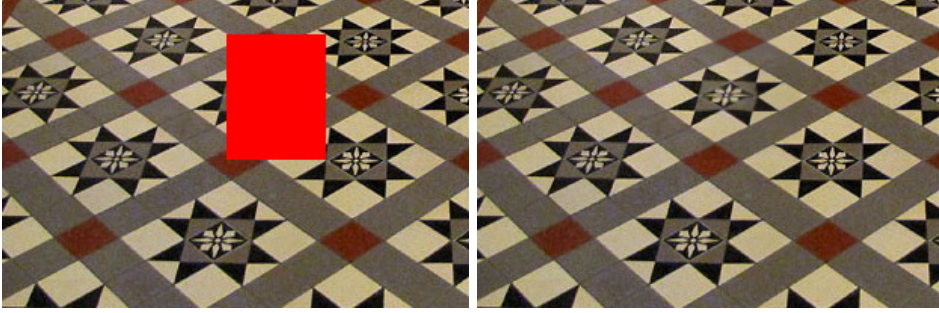


Figure 1: Inpainting of the texture under the perspective distortion. On the left: the original image with the inpainting domain shown in red. On the right: the result produced by our inpainting method.

they have fixed shapes and sizes, defined beforehand. In this thesis we explore the topic of image comparison with shape-adaptive patches that might change their appearance to match the underlying local image content. Consider a simple example, when we want to compare a pair of images, one of which is just a rotated version of another. If we somehow adjust orientations of patches around their centers in accordance with the global rotation, we can obtain a patch similarity measure that is invariant to that rotation and use it to analyze this pair of images properly. In this work we focus on a much more complex case, when patches can be related by *local affine transformations* and a similarity measure thus should be affine invariant in this sense. An affine invariant patch similarity measure based on shape-adaptive patches allows, for instance, to detect similar texture patterns that are distorted differently by the perspective effect, or to compare two images of the same 3D scene, taken from different viewpoints.

This thesis is divided into two parts. In the first part we derive and study two affine invariant similarity measures. For that we consider a recently proposed axiomatic framework for similarity measures between images defined on Riemannian manifolds (Ballester et al. (2014)). From this framework we extract one particular linear model describing a family of similarity measures. The model is represented by a partial differential equation, and we obtain two specific similarity measures as approximate solutions of it. We also propose an iterative construction scheme to define and compute the affine covariant structure tensors that can be used as Riemannian metrics with these similarity measures. We pay a lot of attention to the efficient numerical implementation of the proposed similarity measures. Even though

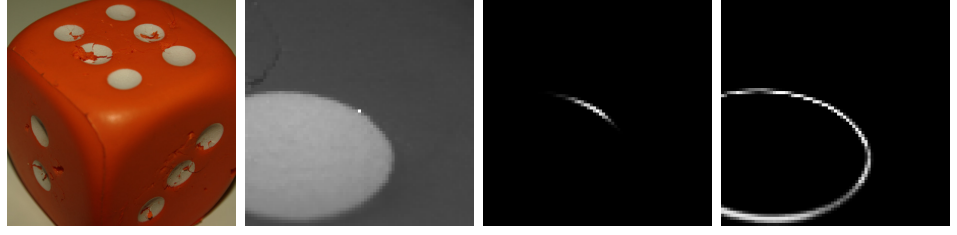


Figure 2: Motivation for the denoising with shape-adaptive patches. From left to right: the original image, the zoom of the search window with its central (reference) point shown in white, similarity values computed between the central point and all other points using the square patches, and using the proposed affine invariant patch similarity measure with shape-adaptive patches. Brighter colors encode higher similarities.

throughout this work we focus only on the 2D images, the proposed affine invariant similarity measures can as well be used for the comparison of videos and 3D images. Therefore, we cover implementation details for both 2D and 3D cases. Finally, we study the proposed similarity measures in the set of experiments.

In the second part of the thesis we present two novel image processing methods that are built around the affine invariant similarity measure, namely an exemplar-based image inpainting method and an image denoising method. Image inpainting, also called image completion, is aimed at the recovery of occluded, missing or corrupted parts of an image, collectively called *inpainting domain*, in such a way that the reconstructed image looks natural. Exemplar-based inpainting methods exploit the self-similarity of natural images by assuming that the missing information can be found elsewhere outside the inpainting domain. The affine invariant similarity measure allows us to properly compare shape-adaptive patches inside and outside the inpainting domain, even if they are related by some local affine transformation. The most similar source patches from the outside can then be copied into the inpainting domain after applying appropriate transformations. In other words, the affine invariant similarity measure effectively extends the space of available source patches. Figure 1 shows an example of the inpainting result obtained using our method for a texture pattern under the perspective effect.

For the image denoising problem we extend the well-known Non-Local Means denoising method using the affine invariant similarity measure. In

the Non-Local Means approach it is assumed that for every noisy reference patch a bunch of other similar noisy patches can be collected together and averaged to produce a single denoised patch. The bigger amount of patches is aggregated, the more noise can be suppressed by averaging. Therefore, the performance of the Non-Local Means denoising is largely determined by the ability to find many similar patches. With the help of the proposed similarity measure we can find and aggregate more similar patches, which in turn leads to better denoising results. This is illustrated in Figure 2. Similarity values are computed between the central (reference) point and all other points within a relatively small window. As can be seen, more points are recognized as similar to the reference in the case when the affine invariant similarity measure is used.

For both presented methods we explain the most important details of their numerical implementation and perform comparisons with other relevant methods from the literature.

Bibliography of the thesis

The following research articles are related to this thesis:

Submitted

- V. Fedorov and C. Ballester. Affine non-local means image denoising. *IEEE Transactions On Image Processing*, 2016. Submitted

Published

- V. Fedorov, P. Arias, G. Facciolo, and C. Ballester. Affine invariant self-similarity for exemplar-based inpainting. In *Proceedings of the 11th Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*, pages 48–58, 2016. doi: 10.5220/0005728100480058
- V. Fedorov, P. Arias, R. Sadek, G. Facciolo, and C. Ballester. Linear multiscale analysis of similarities between images on riemannian manifolds: Practical formula and affine covariant metrics. *SIAM Journal on Imaging Sciences*, 8(3):2021–2069, 2015a. doi: 10.1137/141000002
- V. Fedorov, G. Facciolo, and P. Arias. Variational framework for non-local inpainting. *Image Processing On Line*, 5:362–386, 2015b. doi: 10.5201/ipol.2015.136

Contents

Abstract	ix
Preface	xi
I Patch Similarity Measures	1
1 Introduction	3
2 Preliminaries	9
2.1 Notation	10
2.2 A Priori Connections on Product Manifold	13
3 Axiomatic Approach to Patch Similarity Measures	19
3.1 Axioms	20
3.2 Similarity Measures as Solutions of PDEs	23
4 Derivation of Patch Similarity Measures	25
4.1 Linearity Assumption	25
4.2 The Case of $(\mathcal{M}^k, g^k) = (\mathbb{R}^N, g^k)$	26
4.3 Similarity Measure with Constant Metrics	28
4.4 Approximate Solution for Varying Metrics	29
5 Structure Tensor as a Metric	35
5.1 Affine Covariant Structure Tensors	36
5.2 A Priori Connection from Structure Tensors	41

5.3	Structure Tensors in Discrete Images	46
5.4	Empirical Study of the Construction Scheme	47
5.5	Related Approaches to Affine Covariant Regions	52
6	Numerical Implementation	55
6.1	Outline of Patch Similarity Computation	56
6.2	Affine Covariant Regions	58
6.3	Dominant Orientations	63
6.4	Normalization and Interpolation	67
7	Experimental Results for the Similarity Measures	73
7.1	Multiscale Property	74
7.2	Affine Invariance Property	85
8	Conclusions	93
II	Applications	95
9	Introduction	97
9.1	An Affine Invariant Similarity Measure	97
10	Image Inpainting	103
10.1	Problem Statement	103
10.2	Related Work	105
10.3	Inpainting Formulation	106
10.4	Numerical Implementation	110
10.5	Experimental Results	114
10.6	Conclusions	118
11	Image Denoising	121
11.1	Problem Statement	121
11.2	Related Work	122
11.3	Affine Non-Local Means Denoising	124
11.4	Implementation Details	131
11.5	Parameters Selection	133
11.6	Experimental Results and Assessment	135
11.7	Conclusions	138
12	Conclusions	145

III Appendices	147
A Proof of Theorem 4.1	149
Bibliography	155

PART I

Patch Similarity Measures

Introduction

Image comparison is a topic that has received a lot of attention from the image processing and computer vision communities. It is a main ingredient in many applications such as object recognition, stereo vision, image registration, image denoising, exemplar-based image inpainting, to name a few. There are plenty of techniques for image comparison and the particular choice always depends on the specific task. In some sense the most simple examples of image comparison are the quantitative visual quality assessment measures such as mean square error (MSE), peak signal to noise ratio (PSNR) and structural similarity (SSIM) of Wang et al. (2004). They are widely used, for instance, to measure the quality of lossy compression and restoration of images and videos. Similarity measures, that compare two given images as a whole and assign a single similarity values to them, are sometimes called global. Such similarity measures, for example, Mutual Information (MI) of Viola and Wells (1995) and its multiscale extension of Zimmer and Piella (2014), are typically used as cost functions in image registration.

Very frequently it is required to compare two given points in an image or, in more general case, in two different images. Since a single color value of a point is not very descriptive, it is common to use a small neighborhood around that point in the comparison. Traditionally such neighborhoods are called *patches*. A distinctive property of all patch-wise image comparison techniques, sometimes also called template matching, is that they assign a similarity (or distance) value to any given pair of points. In other words, for every single point in one image there is a dense similarity (distance) field associated with another image. Let $u : \Omega_u \subset \mathbb{R}^N \rightarrow \mathbb{R}^M$ and $v : \Omega_v \subset \mathbb{R}^N \rightarrow$

\mathbb{R}^M be two images with M color channels. Here Ω_u and Ω_v denote their image domains. This general definition of u and v emphasizes that the image comparison theory deals both with regular 2D images and videos, as well as with 3D images, sequences of 3D images captured over time and other more exotic data structures. The patch-wise image comparison functions can then be generalized as

$$\begin{aligned} \mathcal{D} : \Omega_u \times \Omega_v &\longrightarrow \mathbb{R} \\ (x, y) &\longrightarrow \mathcal{D}(x, y) \end{aligned}$$

where u and v may coincide. Let p denote a patch, which usually is given by a connected subset of \mathbb{R}^N centered at the origin. Some examples of such functions, that are usually called *similarity measures*, are:

maximum absolute difference

$$\mathcal{D}_{\max}(x, y) = \max_{h \in p} |u(x + h) - v(y + h)|,$$

Cross-Correlation

$$\mathcal{D}_{\text{CC}}(x, y) = \sum_{h \in p} u(x + h)v(y + h),$$

Sum of Absolute Differences

$$\mathcal{D}_{\text{SAD}}(x, y) = \sum_{h \in p} |u(x + h) - v(y + h)|,$$

Sum of Squared Differences

$$\mathcal{D}_{\text{SSD}}(x, y) = \sum_{h \in p} (u(x + h) - v(y + h))^2.$$

Of course, much more elaborated similarity measures can be found in the literature, such as Zero-mean Normalized Cross-Correlation (ZNCC) in [Lewis \(1995\)](#), Phase Correlation in [Foroosh et al. \(2002\)](#), IMage Euclidean Distance (IMED) in [Wang et al. \(2005\)](#), etc.

In some contexts, for example, in object detection and object recognition, it is common to compute similarity values sparsely by detecting keypoints or salient regions (collectively called *features*) in the images being compared. Detected features are then described in one way or another, and a matching cost is assigned to a given pair of descriptors. For such applications it is much more important to have similarity values (or matching costs) at a

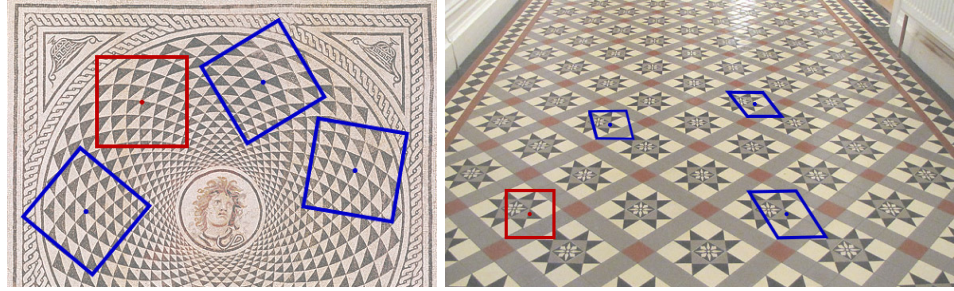


Figure 1.1: Extension of the space of available patches. On the left: rotations. On the right: affine transformations.

few repetitive and distinctive locations, rather than to be able to compute them densely. This freedom of choosing the features to work with leads to the image comparison that can be invariant to different kinds of image transformations, or at least sufficiently invariant to be used in numerous applications. For example, multiple feature detectors and descriptors, that are rotation and scale invariant, were developed recently (see Tuytelaars and Mikolajczyk (2008)). Among them SIFT of Lowe (2004) is probably the most widely known. The main idea behind these approaches is to build a scale-space of an image by low-pass filtering and then to search through this scale-space for scale invariant keypoints. Some methods go beyond that and allow to detect and describe features in an affine invariant way, for example, ASIFT of Morel and Yu (2009) or Harris-Affine of Mikolajczyk and Schmid (2004).

In this work we focus on the image comparison by the means of dense patch-wise similarity measures. We consider patches to be the basic units of information that allow us to analyze and exploit the self-similarity property, usually attributed to natural images. Commonly, patches have square shape and fixed size, and are used as they appear in an image without any transformations. However, in the image formation process final appearance of visual details of an observed scene is affected by the geometry of that scene and by their positions with respect to the camera. Therefore, it might be beneficial to consider shape-adaptive patches despite the additional computational burden associated with them. This point is illustrated by Figure 1.1, where reference patches are shown in red and several other patches, containing similar visual details, are shown in blue. In the left image similar patches are related by rotations. In the right image patches

containing the same pieces of texture are related by affine transformations. Some examples of image processing methods that exploit shape-adaptive patches are Foi et al. (2007); Deledalle et al. (2012). Our goal is to introduce and study a family of affine invariant similarity measures that quite naturally also imply shape-adaptive patches. These similarity measures provide dense image comparison given an arbitrary pair of points, which at the same time is invariant to affine transformations. Potential applications of the affine invariant patch similarity measures are numerous, in particular, in Part II we present two novel methods for image denoising and image inpainting.

In order to derive the affine invariant similarity measures we exploit an axiomatic approach that previously was used in Alvarez et al. (1993) to classify and study multiscale analyses of images. *Multiscale analysis of images* was defined there as a family of transforms T_t ($t \geq 0$) which, when applied to an image u , yield a sequence of images $u(t) = T_t(u)$ at different scales. This is a classical technique in image processing and computer vision and such sequence of images is usually called the *scale-space*. The scale is related to the degree of smoothing or to the size of the neighborhood which is used to give an estimate of the brightness of the picture at a given point. The core idea of the axiomatic approach is that image processing transforms in general must satisfy some formal requirements which can be represented by a set of axioms. A particular transform can then be deduced from a given set of axioms. By changing the set of axioms one can obtain transforms with different properties. Therefore, it provides a useful tool for classification of already existing transforms and also for exploration of the new ones. For example, in Alvarez et al. (1993) it was shown that under reasonable assumptions (compositions of axioms) all scale-spaces are (viscosity) solutions of partial differential equations of second order. The classification, given in that work, covers many well-known techniques such as mathematical morphology operations (dilation and erosion), Gaussian scale-space, Perona-Malik diffusion and mean curvature motion. In addition, the Affine Morphological Scale Space (AMSS), which is affine invariant and corresponds to motion of level lines by the power $\frac{1}{3}$ of its curvature, was obtained.

Later on this axiomatic approach was extended for images defined on Riemannian manifolds in Calderero and Caselles (2014). Such manifolds arise, for instance, for images defined on \mathbb{R}^N , endowed with a suitable metric depending on the image. Among other contributions this generalization allowed to include the anisotropic scale-spaces in the common classification.

Recently the same approach was used in Ballester et al. (2014) to classify and study not scale-spaces of images, but rather multiscale similarity measures. It was shown that multiscale analyses of similarities between images on Riemannian manifolds, satisfying a certain set of axioms, are (viscosity) solutions of a family of degenerate PDEs. Interestingly enough, the Sum of Squared Differences discussed above also fits in this framework. This simple similarity measure can be written in the continuous setting as

$$\mathcal{D}(t, x, y) = \int_{\mathbb{R}^2} g_t(h)(u(x+h) - v(y+h))^2 dh, \quad (1.1)$$

where g_t is a given windowing function that we assume to be Gaussian of variance t . Then it can be shown that (1.1) solves

$$\frac{\partial \mathcal{D}}{\partial t} = \Delta_x \mathcal{D} + 2\text{Tr}(D_{xy}^2 \mathcal{D}) + \Delta_y \mathcal{D}, \quad (1.2)$$

which is probably the simplest case of linear PDE expressing the multiscale comparison of two image patches. Notice that we refer to this similarity measure as *multiscale*, because the naturally appearing windowing function g_t allows to control the support (or scale) of patch comparison by adjusting the t parameter.

In our work we study one particular linear model proposed among many others in Ballester et al. (2014). In order to keep this dissertation self-contained, in Chapter 2 we summarize some basic definitions and notation from Tensor Calculus and Differential Geometry and also recall from Ballester et al. (2014) the notion of an *a priori connection* which plays an important role in this work. Then in Chapter 3 we define the particular set of axioms that should be satisfied by the linear model we are interested in. In Chapter 4 we finally present this model which is generic and at the same time leads to computationally feasible similarity measures that have the complexity of the usual Sum of Squared Differences. We should remark at this point, that the PDE corresponding to the linear model involves spatially varying metrics $G_1(x)$ and $G_2(y)$ in image domains

$$\frac{\partial \mathcal{D}}{\partial t} = \text{Tr}(G_1(x)^{-1} D_x^2 \mathcal{D}) + 2\text{Tr}(G_1(x)^{-\frac{1}{2}} G_2(y)^{-\frac{1}{2}} D_{xy}^2 \mathcal{D}) + \text{Tr}(G_2(y)^{-1} D_y^2 \mathcal{D}). \quad (1.3)$$

This means that the computational complexity of solving (1.3) is of order S^4 , if each image is determined on a grid of size S^2 . For that reason we use WKB approximation, named after Wentzel, Kramers and Brillouin, to develop an approximate solution that gives two practical similarity measures.

We also focus on the problem of defining the metrics G_i , $i = 1, 2$, such that the resulting multiscale similarity measure is affine invariant. It is well known that the *structure tensors* can be used as metrics in image domains. In Chapter 5 we propose a novel iterative scheme to define and compute structure tensors which are guaranteed, at least theoretically, to be *affine covariant*. By affine covariance we mean that the structure tensors transform appropriately by affine transformations. We study their properties both theoretically and experimentally. We also show that affine covariant neighborhoods (or shape-adaptive patches) are naturally defined by the affine covariant structure tensors.

We pay much attention to the efficient numerical implementation of the affine invariant similarity measures and thus present the most important details of it in Chapter 6. We cover both 2D and 3D cases which are the most practical ones. In Chapter 7 we present experimental results illustrating multiscale and affine invariance properties of the proposed similarity measures. Finally, Chapter 8 concludes the first part of the thesis.

Preliminaries

In this chapter we collect some basic definitions and notation from Tensor Calculus and Differential Geometry, in particular about Riemannian manifolds, that will be useful for the following theoretical derivations. We also recall the notion of *a priori connection* from [Ballester et al. \(2014\)](#).

Throughout this work we consider images defined on Riemannian manifolds. A subset $\mathcal{M} \subset \mathbb{R}^P$ is called a *smooth N -dimensional manifold* in \mathbb{R}^P ($P, N \in \mathbb{N}_0$ and $P \geq N$), if every point ξ of \mathcal{M} has an open neighborhood $V \subset \mathcal{M}$ that is diffeomorphic to an open subset $U \subset \mathbb{R}^N$. In other words, locally an N -dimensional manifold looks like some Euclidean space \mathbb{R}^N . A diffeomorphism $\phi : V \rightarrow U$ is called a *coordinate chart* of \mathcal{M} and its inverse $\psi := \phi^{-1} : U \rightarrow V$ is called a *parametrization* of \mathcal{M} around point ξ (see Figure 2.1). In principle there is no need for an ambient space \mathbb{R}^P to define a manifold \mathcal{M} .

The simplest example of an N -dimensional manifold is the Euclidean space \mathbb{R}^N itself. A more illustrative example is the smooth 2-dimensional manifold defined as

$$\mathcal{M} := S^2 = \{(x, y, z) \in \mathbb{R}^3 \mid x^2 + y^2 + z^2 = 1\},$$

which is a sphere in \mathbb{R}^3 . Let $V \subset \mathbb{R}^3$ and $U \subset \mathbb{R}^2$ be the open sets

$$V := \{(x, y, z) \in \mathcal{M} \mid z > 0\}, \quad U := \{(x, y) \in \mathbb{R}^2 \mid x^2 + y^2 < 1\}.$$

The map $\phi : V \rightarrow U$ is then given by

$$\phi(x, y, z) := (x, y).$$

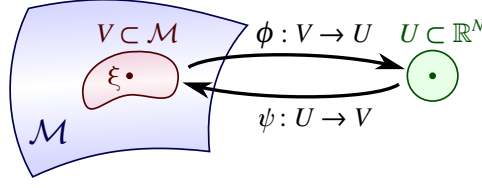


Figure 2.1: Open sets $V \subset \mathcal{M}$ and $U \subset \mathbb{R}^N$, coordinate chart $\phi: V \rightarrow U$ and coordinate system $\psi := \phi^{-1}: U \rightarrow V$.

It is bijective and its inverse $\psi := \phi^{-1}: U \rightarrow V$ is given by

$$\psi(x, y) = (x, y, \sqrt{1 - x^2 - y^2}).$$

Since both ϕ and ψ are smooth, the map ϕ is a coordinate chart on S^2 . Similarly, the open sets corresponding to $z < 0$, $y > 0$, $y < 0$, $x > 0$ and $x < 0$ can be used to completely cover S^2 by six coordinate charts. Hence S^2 is a smooth manifold.

A Riemannian manifold is a smooth (C^∞) manifold equipped with a Riemannian metric, which provides smoothly varying choices of inner products on tangent spaces and allows one to measure geometric quantities such as distances and angles. For instance, the Euclidean metric is an example of Riemannian metric.

2.1 Notation

Let (\mathcal{N}, g) be a smooth Riemannian manifold of dimension N , endowed with a general metric g . Let ξ be a point on \mathcal{N} . Let $U \subseteq \mathbb{R}^N$ be an open set, containing 0, and $\psi: U \rightarrow \mathcal{N}$ be any coordinate system such that $\psi(0) = \xi$. Let $g_{ij}(\xi)$ and $\Gamma_{ij}^{\mathcal{N},k}(\xi)$ (indices i, j, k run from 1 to N) denote, respectively, the coefficients of the first fundamental form (metric tensor) of \mathcal{N} and the Christoffel symbols, computed in the coordinate system ψ around ξ . For simplicity we shall denote by $G(\xi)$ the symmetric matrix $[g_{ij}(\xi)]$ and by $\Gamma^{\mathcal{N},k}(\xi)$ the matrix formed by the coefficients $[\Gamma_{ij}^{\mathcal{N},k}(\xi)]$, $i, j = 1, \dots, N$ for each $k = 1, \dots, N$.

We shall use the Einstein summation convention that implies summation over repeated indices. That is, if otherwise is not stated, $a_i b^i$ represents $\sum_i a_i b^i$ and superscript indices should not be confused with exponents.

As usual, given a point $\xi \in \mathcal{N}$, we denote by $T_\xi \mathcal{N}$ the tangent space to \mathcal{N} at the point ξ . By $T_\xi^* \mathcal{N}$ we denote its dual space. The scalar product of two vectors $v, w \in T_\xi \mathcal{N}$ is denoted by $\langle v, w \rangle_\xi$ and, in the coordinate system $\psi : U \rightarrow \mathcal{N}$, is computed as $\langle v, w \rangle_\xi = g_{ij}(\xi) v^i w^j$, where v^i, w^i are the contravariant coordinates of v and w in the basis $\frac{\partial}{\partial x^i}|_\xi$ of $T_\xi \mathcal{N}$. The action of a covector $p^* \in T_\xi^* \mathcal{N}$, on a vector $v \in T_\xi \mathcal{N}$, is denoted by $(p^*, v) = p_i v^i$. Here p_i denotes covariant coordinates of p . The relation between contravariant and covariant coordinates is given by

$$p_i = g_{ij}(\xi) p^j \quad \text{and} \quad p^i = g^{ij}(\xi) p_j, \quad (2.1)$$

where $g^{ij}(\xi)$ denotes the coefficients of the inverse matrix of $g_{ij}(\xi)$. We can also write (2.1) as

$$p^* = G(\xi) p \quad \text{and} \quad p = G^{-1}(\xi) p^*.$$

In this way $G(\xi) : T_\xi \mathcal{N} \rightarrow T_\xi^* \mathcal{N}$. Notice that we may also write $(p^*, v) = g_{ij}(\xi) p^j v^i$.

In the case when ψ is a geodesic normal coordinate system centered at ξ , the matrix $G(\xi)$ is the identity matrix $I = (\delta_{ij})$, and I maps vectors to covectors, that is, $I : T_\xi \mathcal{N} \rightarrow T_\xi^* \mathcal{N}$ (with the same coordinates in the dual basis). We shall denote by I^{-1} the inverse of I , mapping covectors to vectors.

Maps and Symmetric Maps

We shall use the coordinate system ψ to express a bilinear map $\hat{A} : T_\xi \mathcal{N} \times T_\xi \mathcal{N} \rightarrow \mathbb{R}$. Indeed, if $[A_{ij}]$ is the matrix of \hat{A} in this basis, and $v, w \in T_\xi \mathcal{N}$, we may write $\hat{A}(v, w) = A_{ij} v^j w^i$. If $A_j^i = g^{ik}(\xi) A_{kj}$, then $[A_j^i]$ determines a map $A : T_\xi \mathcal{N} \rightarrow T_\xi \mathcal{N}$ such that $\hat{A}(v, w) = \langle Av, w \rangle = (G(\xi) Av, w)$. Observe that $G(\xi) A : T_\xi \mathcal{N} \rightarrow T_\xi^* \mathcal{N}$. Observe also that our notation A_j^i already indicates that $A = [A_j^i]$ maps vectors to vectors. In our notation, we shall not distinguish between matrices and maps.

As usual, we say that a linear map $L : T_\xi \mathcal{N} \rightarrow T_\xi^* \mathcal{N}$ is symmetric if $(Lv, w) = (Lw, v)$ for any $v \in T_\xi \mathcal{N}$, $w \in T_\xi \mathcal{N}$. We shall use the notation

$$SM_\xi(\mathcal{N}) := \{A : T_\xi \mathcal{N} \rightarrow T_\xi^* \mathcal{N}, A \text{ is symmetric}\}.$$

We shall also write

$$S_\xi(\mathcal{N}) := \{A : T_\xi \mathcal{N} \rightarrow T_\xi \mathcal{N}, G(\xi) A \in SM_\xi(\mathcal{N})\}.$$

From now on, when the point $\xi \in \mathcal{N}$ is understood, we write G instead of $G(\xi)$.

Notice that if $A : T_\xi \mathcal{N} \rightarrow T_\xi \mathcal{N}$, we define $A^t : T_\xi^* \mathcal{N} \rightarrow T_\xi^* \mathcal{N}$ by

$$(A^t p, v) = (p, Av) \quad \forall v \in T_\xi \mathcal{N}, p \in T_\xi^* \mathcal{N}.$$

This equality will be useful later on, in the proof of the Theorem 4.1.

Rotations in the Tangent Space

We define a rotation $R : T_\xi \mathcal{N} \rightarrow T_\xi \mathcal{N}$ as a linear map that satisfies

$$\langle Rv, Rw \rangle = \langle v, w \rangle \quad \forall v, w \in T_\xi \mathcal{N}.$$

Notice that R does not affect lengths and angles of vectors in the tangent space, hence it satisfies

$$R^t G R = G.$$

Let $B : T_\xi \mathcal{N} \rightarrow T_\xi \mathcal{N}$ be a matrix such that $BI^{-1}B^t = G^{-1}$. Thus, $B^t G B = I$ and B is mapping an orthonormal basis of $(T_\xi \mathcal{N}, I)$ to an orthonormal basis of $(T_\xi \mathcal{N}, G(\xi))$.

If $R : T_\xi \mathcal{N} \rightarrow T_\xi \mathcal{N}$ is a rotation then

$$(B^{-1}RB)^t I (B^{-1}RB) = I.$$

That is, $B^{-1}RB$ is a classical rotation.

Gradient and Hessian

Given a function u on \mathcal{N} , we denote by $D_{\mathcal{N}}u$ and $D_{\mathcal{N}}^2u$ the gradient and Hessian of u , respectively. In a coordinate system, $D_{\mathcal{N}}u$ is the covector $\frac{\partial u}{\partial x^i}$, and $D_{\mathcal{N}}^2u$ is the matrix $\frac{\partial^2 u}{\partial x^i \partial x^j} - \Gamma_{ij}^{\mathcal{N},k} \frac{\partial u}{\partial x^k}$ which acts on tangent vectors. Therefore, in this notation $D_{\mathcal{N}}^2u(\xi) : T_\xi \mathcal{N} \times T_\xi \mathcal{N} \rightarrow \mathbb{R}$ is a bilinear map, and is a symmetric matrix in coordinates. We denote by $\nabla_{\mathcal{N}}u$ the vector of coordinates $g^{ij} \frac{\partial u}{\partial x^j}$. Then, $|\nabla_{\mathcal{N}}u(\xi)|_\xi^2 = \langle \nabla_{\mathcal{N}}u(\xi), \nabla_{\mathcal{N}}u(\xi) \rangle_\xi$. To simplify the notation we shall write Du and ∇u instead of $D_{\mathcal{N}}u$, and $\nabla_{\mathcal{N}}u$. The vector field ∇u satisfies $\langle \nabla u, v \rangle_\xi = du(v)$, $v \in T_\xi \mathcal{N}$, du being the differential of u .

Product Manifold $\mathcal{N} = \mathcal{M}^1 \times \mathcal{M}^2$

Let (\mathcal{M}^i, g^i) be a smooth Riemannian manifold with metric g^i , for $i = 1, 2$. Let $\Gamma^{(i)}$ be the connection on \mathcal{M}^i . We shall work here with a product manifold $\mathcal{N} = \mathcal{M}^1 \times \mathcal{M}^2$ with the metric $g = g^1 \times g^2$, so that $T_\xi \mathcal{N} = T_{\xi_1} \mathcal{M}^1 \times T_{\xi_2} \mathcal{M}^2$, $\xi = (\xi_1, \xi_2) \in \mathcal{M}^1 \times \mathcal{M}^2$. If $(v_i, w_i) \in T_{\xi_1} \mathcal{M}^1 \times T_{\xi_2} \mathcal{M}^2$, $\xi = (\xi_1, \xi_2) \in \mathcal{M}^1 \times \mathcal{M}^2$, then we consider the metric

$$\langle (v_1, w_1), (v_2, w_2) \rangle_\xi = \langle v_1, v_2 \rangle_{\xi_1} + \langle w_1, w_2 \rangle_{\xi_2} = (G^1(\xi_1)v_1, v_2) + (G^2(\xi_2)w_1, w_2).$$

With a slight abuse of notation, let us write $G(\xi) = \text{diag}(G^1(\xi_1), G^2(\xi_2))$.

Let $\xi = (\xi_1, \xi_2) \in \mathcal{M}^1 \times \mathcal{M}^2$. We consider a coordinate system of the form $\psi = (\psi_1, \psi_2) : U_1 \times U_2 \rightarrow \mathcal{M}^1 \times \mathcal{M}^2$ with $\psi_i(0) = \xi_i$, U_i being a neighborhood of 0 in \mathbb{R}^N . We also consider points $x \in U_1$ and $y \in U_2$.

Without loss of generality, let us assume that \mathcal{M}^1 and \mathcal{M}^2 have dimension N . Let us denote the connection on $\mathcal{M}^1 \times \mathcal{M}^2$ as $\Gamma := \Gamma^{(1)} \otimes \Gamma^{(2)}$ with indices $i, j, k \in \{1, \dots, 2N\}$, with $\xi_i = \xi_{1i}$, $i \in \{1, \dots, N\}$, and $\xi_i = \xi_{2(i-N)}$, $i \in \{N+1, \dots, 2N\}$. We can denote the coordinates as z^i , $i \in \{1, \dots, 2N\}$ with $z^i = x^i$, $i \in \{1, \dots, N\}$, and $z^i = y^{i-N}$, $i \in \{N+1, \dots, 2N\}$. Using the formula

$$(\Gamma^{(1)} \otimes \Gamma^{(2)})_{ij}^k = \frac{1}{2} g^{kl} \left(\frac{\partial g_{jl}}{\partial z^i} + \frac{\partial g_{il}}{\partial z^j} - \frac{\partial g_{ij}}{\partial z^l} \right),$$

we obtain

$$(\Gamma^{(1)} \otimes \Gamma^{(2)})^k(x, y) = \begin{bmatrix} \Gamma^{(1)k}(x) & 0 \\ 0 & \Gamma^{(2)k}(y) \end{bmatrix}.$$

We denote by $SM_\xi(\mathcal{N})$ the set of symmetric matrices of size $2N \times 2N$ in $\mathcal{N} = \mathcal{M}^1 \times \mathcal{M}^2$.

2.2 A Priori Connections on Product Manifold

The concept of a *a priori connection* is important for this work and we need to clarify it. Suppose that both manifolds \mathcal{M}^1 and \mathcal{M}^2 coincide with \mathbb{R}^N endowed with the Euclidean metric. Let u and v be two given images defined on \mathbb{R}^N . Then it would be standard to use the L^2 distance to compare the patches centered at x and y ,

$$\mathcal{D}(t, x, y) = \int_{\mathbb{R}^N} g_t(h) (u(x+h) - v(y+h))^2 dh, \quad (2.2)$$

where g_t is a given window that we assume to be Gaussian of variance t . But if the image v is rotated we could also use the L^2 distance between u and a rotated version of v (around y), namely,

$$\mathcal{D}(t, x, y) = \int_{\mathbb{R}^N} g_t(h) (u(x+h) - v(y+Rh))^2 dh. \quad (2.3)$$

We admit that this decision is taken a priori and is done thanks to an operator that connects the tangent plane at both points.

Definition 2.1. *We say that $P(\xi)$, $\xi = (\xi_1, \xi_2) \in \mathcal{N}$, is an a priori connection map in \mathcal{N} if $P(\xi) : (T_{\xi_1}\mathcal{M}^1, G^1(\xi_1)) \rightarrow (T_{\xi_2}\mathcal{M}^2, G^2(\xi_2))$ is an isometry, that is,*

$$\langle P(\xi)v, P(\xi)w \rangle_{G^2(\xi_2)} = \langle v, w \rangle_{G^1(\xi_1)} \quad \forall v, w \in T_{\xi_1}\mathcal{M},$$

and we assume also that the map is differentiable in ξ .

Given an a priori connection $P(\xi) : (T_{\xi_1}\mathcal{M}^1, G^1(\xi_1)) \rightarrow (T_{\xi_2}\mathcal{M}^2, G^2(\xi_2))$, we can also define its inverse $P(\xi)^{-1} : (T_{\xi_2}\mathcal{M}^2, G^2(\xi_2)) \rightarrow (T_{\xi_1}\mathcal{M}^1, G^1(\xi_1))$. For simplicity, and because the arguments in P clearly specify whether it goes from \mathcal{M}^1 to \mathcal{M}^2 or inversely, we denote $P(\xi_2, \xi_1) = P(\xi_1, \xi_2)^{-1}$, so that we have

$$P(\xi_2, \xi_1)P(\xi_1, \xi_2) = I. \quad (2.4)$$

Remark 1. Note that if we have a complete manifold with empty cut locus, we can define the a priori connection in it by parallel transport without ambiguities.

Remark 2. Let $\text{Isom}(T\mathcal{M}^i)$ denote the set of isometry maps in the tangent bundle $T\mathcal{M}^i$. Let $R^1 : \mathcal{M}^1 \rightarrow \text{Isom}(T\mathcal{M}^1)$ and $R^2 : \mathcal{M}^2 \rightarrow \text{Isom}(T\mathcal{M}^2)$ be two given maps. If $P(\xi)$ is an a priori connection, then $R^2(\xi_2)P(\xi)R^1(\xi_1)$ is also an a priori connection.

The isometry property can be written as

$$(P(\xi)^t G^2(\xi_2) P(\xi) v, w) = (G^1(\xi_1) v, w),$$

where $P(\xi)$ is expressed in the basis of $T_{\xi_1}\mathcal{M}^1$ associated with the metric $G^1(\xi_1)$ and the basis of $T_{\xi_2}\mathcal{M}^2$ associated with the metric $G^2(\xi_2)$. Then

$$P(\xi)^t G^2(\xi_2) P(\xi) = G^1(\xi_1). \quad (2.5)$$

Note that from (2.5) it also follows that, for smooth and orientable manifolds, an a priori connection can be found when the metrics are known.

A Priori Connections in Different Coordinate Systems

Let us recall from Ballester et al. (2014) how to compute the a priori connection in another coordinate system. The map $P(\xi)$ expresses the a priori connection in the coordinate system $\psi_1 \rightarrow \psi_2$. Let $\bar{\psi} = (\bar{\psi}_1, \bar{\psi}_2)$ be another coordinate system around ξ . Let $G^i(\xi_i)$ and $\bar{G}^i(\xi_i)$, $i = 1, 2$, be the metric matrices represented in the coordinate systems ψ_i and $\bar{\psi}_i$, respectively. Let $B_{G^i, \bar{G}^i}(\xi_i) = D(\psi_i^{-1} \circ \bar{\psi}_i)(0)$, $i = 1, 2$, and $B_{G, \bar{G}}(\xi) = (B_{G^1, \bar{G}^1}(\xi_1), B_{G^2, \bar{G}^2}(\xi_2))$. Note that

$$B_{G^i, \bar{G}^i}(\xi_i) : (T_{\xi_i} \mathcal{M}^i, \bar{G}^i(\xi_i)) \rightarrow (T_{\xi_i} \mathcal{M}^i, G^i(\xi_i))$$

is such that $B_{G^i, \bar{G}^i}(\xi_i)^t G^i(\xi_i) B_{G^i, \bar{G}^i}(\xi_i) = \bar{G}^i(\xi_i)$. Note also that all matrices here are uniquely defined. Using the last equality, (2.5) can be expressed as

$$\begin{aligned} P^t(\xi) B_{G^2, \bar{G}^2}(\xi_2)^{-t} \bar{G}^2(\xi_2) B_{G^2, \bar{G}^2}(\xi_2)^{-1} P(\xi) = \\ B_{G^1, \bar{G}^1}(\xi_1)^{-t} \bar{G}^1(\xi_1) B_{G^1, \bar{G}^1}(\xi_1)^{-1}. \end{aligned} \quad (2.6)$$

If we define

$$\bar{P}(\xi) := B_{G^2, \bar{G}^2}(\xi_2)^{-1} P(\xi) B_{G^1, \bar{G}^1}(\xi_1), \quad (2.7)$$

then $\bar{P}(\xi)$ is an a priori connection in the coordinate system $\bar{\psi}_1 \rightarrow \bar{\psi}_2$, $\bar{P}(\xi) : (T_{\xi_1} \mathcal{M}^1, \bar{G}^1(\xi_1)) \rightarrow (T_{\xi_2} \mathcal{M}^2, \bar{G}^2(\xi_2))$. Indeed, we can express (2.6) as

$$\bar{P}(\xi)^t \bar{G}^2(\xi_2) \bar{P}(\xi) = \bar{G}^1(\xi_1),$$

which is the isometry property defining a priori connections. We say that $\bar{P}(\xi)$ is the derived a priori connection from $P(\xi)$ and $\bar{\psi}$.

Then, (2.7) can be re-written as

$$B_{G^2, \bar{G}^2}(\xi_2) \bar{P}(\xi) = P(\xi) B_{G^1, \bar{G}^1}(\xi_1), \quad (2.8)$$

and we see that both maps $B_{G^1, \bar{G}^1}(\xi_1)$ and $B_{G^2, \bar{G}^2}(\xi_2)$ reflect the same rotation when expressed in the corresponding a priori connections $P(\xi)$ and $\bar{P}(\xi)$, respectively.

Definition 2.2. We say that the coordinate systems $\psi, \bar{\psi}$ are $P(\xi)$ -related if $\bar{P}(\xi)$ is defined by (2.7).

Let us consider the case where $\mathcal{M}^1 = \mathcal{M}^2 = \mathcal{M}$ and $P(\xi)$ as an internal a priori connection given from parallel transport between ξ_1 and ξ_2 , which is an isometry. Then one can define $\bar{P}(\xi)$ by parallel transport expressed in the coordinate systems $\bar{\psi}_1, \bar{\psi}_2$.

Generation of A Priori Connections

In order to generate an a priori connection in a given coordinate system, we first fix a geodesic coordinate system around each point of \mathcal{M}^i . In this coordinate system $I^i(\xi_i)$ denotes the metric for every point $\xi_i \in \mathcal{M}^i$. For each $\xi \in \mathcal{N}$, consider an isometry map (assuming that it exists)

$$Q(\xi) : (T_{\xi_1}\mathcal{M}^1, I^1(\xi_1)) \rightarrow (T_{\xi_2}\mathcal{M}^2, I^2(\xi_2)).$$

Let $\text{Isom}((T\mathcal{M}^1, I^1), (T\mathcal{M}^2, I^2))$ denote the set of such maps. Notice that this is nothing else than an a priori connection – we just express the same concept in different coordinate systems. Now, using the transformation of a priori connections discussed above, we can take an a priori connection Q in a geodesic coordinate field \mathcal{GS} and derive its expression in another coordinate system field.

Let $B^i(\xi_i) : (T_{\xi_i}\mathcal{M}^i, I^i(\xi_i)) \rightarrow (T_{\xi_i}\mathcal{M}^i, G^i(\xi_i))$ be the corresponding canonical maps connecting a geodesic coordinate system \mathcal{GS} around points ξ_i to $(T_{\xi_i}\mathcal{M}^i, G^i(\xi_i))$. Thus,

$$B^i(\xi_i)^t G^i(\xi_i) B^i(\xi_i) = I^i(\xi_i).$$

Note that the map $B^i(\xi_i)$ is uniquely defined by the coordinate systems. When changing, for example, rotating the geodesic coordinate system we get a different matrix.

Let $Q(\xi) \in \text{Isom}((T\mathcal{M}^1, I^1), (T\mathcal{M}^2, I^2))$, where each I is referred to \mathcal{GS} , and let us define

$$P(\xi) := B^2(\xi_2)Q(\xi)B^1(\xi_1)^{-1}. \quad (2.9)$$

Then $P(\xi)$ is an a priori connection map (see Figure 2.2).

Let us note that for any $B^i(\xi_i)$, defined by the choice of new coordinate systems, there are infinitely many a priori connections. Indeed, the initial isometry map $Q(\xi)$ in (2.9) may well be any arbitrary orthogonal transformation (rotation and/or reflection). This will become important later on, and in Section 5.2 we will propose a practical approach to calculate a unique a priori connection.

Related Rotations

Let us consider a coordinate system field and an a priori connection $P(\xi) : (T_{\xi_1}\mathcal{M}^1, G^1(\xi_1)) \rightarrow (T_{\xi_2}\mathcal{M}^2, G^2(\xi_2))$ in that system field. Let us consider a second coordinate system field with metric $\overline{G}^i(\xi_i) = G^i(\xi_i)$, $i = 1, 2$, for each

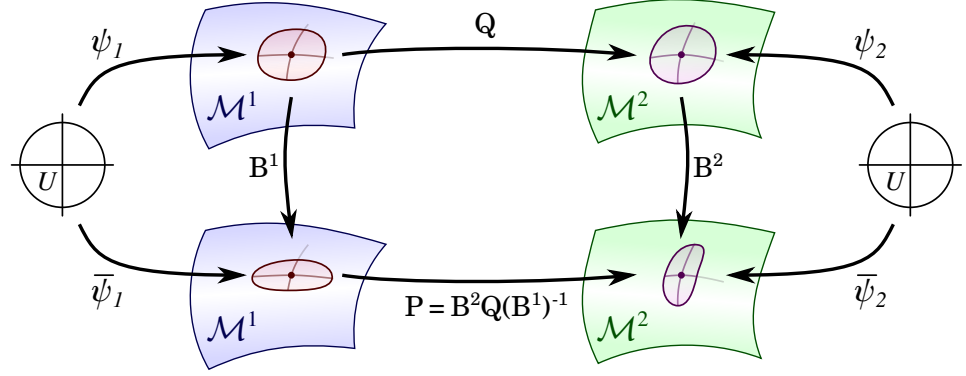


Figure 2.2: Schematic illustration of generation of an a priori connection.

$\xi_i \in \mathcal{M}^i$ so that $B_{G^i, \bar{G}^i}(\xi_i)$ is an isometry field. Let $R^i(\xi_i) := B_{G^i, \bar{G}^i}(\xi_i)$. Let $\bar{P}(\xi)$ be the derived connection. Then (2.8) can be written as

$$R^2(\xi_2) = P(\xi) R^1(\xi_1) \bar{P}(\xi)^{-1}. \quad (2.10)$$

We say that $(R^1(\xi_1), R^2(\xi_2))$ are $P(\xi)$ -related. We call $R = (R^1(\xi_1), R^2(\xi_2))$ a diagonally related rotation (or just a diagonal rotation, if no confusion arises).

Related Germs of Functions on $\mathcal{N} = \mathcal{M}^1 \times \mathcal{M}^2$

Finally, we introduce the space of similarity measures. Let $C_b(\mathcal{N})$ denote the space of bounded continuous functions in \mathcal{N} with the maximum norm. We think of $C_b(\mathcal{N})$ as the space of similarity functions on $\mathcal{N} = \mathcal{M}^1 \times \mathcal{M}^2$. We denote by $C_b^\infty(\mathcal{N})$ the space of infinitely differentiable functions on \mathcal{N} .

Let $C \in C_b(\mathcal{N})$. Let us denote

$$(C, \psi)(x, y) = C(\psi_1(x), \psi_2(y)), \quad \forall (x, y) \in U_1 \times U_2.$$

We can say that $\psi = (\psi_1, \psi_2)$ and $\bar{\psi} = (\bar{\psi}_1, \bar{\psi}_2)$ are $P(\xi)$ -related, if (2.10) holds. If $\bar{\psi}$ is $P(\xi)$ -related to ψ , we write $(C, \bar{\psi})$ as $R(C, \psi)$. Note that $R(C, \psi)$ is a linear map for the restriction of functions in $C_b(\mathcal{N})$ to a neighborhood of $(0, 0)$.

Let us finally introduce notations for the gradient and the Hessian of C , namely,

$$D_{\mathcal{N}}C = (D_x C, D_y C) \quad \text{and} \quad D_{\mathcal{N}}^2 C = \begin{bmatrix} D_{\mathcal{N},xx}C & D_{\mathcal{N},xy}C \\ D_{\mathcal{N},xy}C & D_{\mathcal{N},yy}C \end{bmatrix}.$$

In coordinates, with $i, j, k \in \{1, \dots, N\}$,

$$D_{\mathcal{N}}^2 C = \begin{bmatrix} \frac{\partial^2 C}{\partial x^i \partial x^j} & \frac{\partial^2 C}{\partial x^i \partial y^j} \\ \frac{\partial^2 C}{\partial y^j \partial x^i} & \frac{\partial^2 C}{\partial y^i \partial y^j} \end{bmatrix} - \begin{bmatrix} \Gamma^{(1)k}(x) \frac{\partial C}{\partial x^k} & 0 \\ 0 & \Gamma^{(2)k}(y) \frac{\partial C}{\partial y^k} \end{bmatrix}.$$

Axiomatic Approach to Patch Similarity Measures

In this chapter we describe an axiomatic approach that allows one to derive various patch similarity measures that differ in their properties. A similar approach was originally proposed in [Alvarez et al. \(1993\)](#) for classification and study of multiscale analyses of images. “Multiscale analysis” was defined there as a family of transforms T_t ($t \geq 0$) which, when applied to an image u , yield a sequence of images $u(t) = T_t(u)$ at different scales. The scale is related to the degree of smoothing or to the size of the neighborhood which is used to give an estimate of the brightness of the picture at a given point. Their classification covers many well-known techniques including the Gaussian scale-space ([Marr and Hildreth \(1980\)](#); [Koenderink \(1984\)](#); [Koenderink and Van Doorn \(1986\)](#); [Witkin \(1984\)](#); [Hummel et al. \(1985\)](#); [Hummel \(1987\)](#); [Lindeberg \(1993\)](#); [Weickert et al. \(1999\)](#)), Perona-Malik diffusion ([Perona and Malik \(1990a\)](#); [Catté et al. \(1992\)](#)), mean curvature motion ([Alvarez et al. \(1992\)](#)), mathematical morphology operations (dilation and erosion), the affine morphological scale-space ([Alvarez et al. \(1993\)](#); [Sapiro and Tannenbaum \(1993, 1994\)](#); [Olver et al. \(1993\)](#); [Guichard and Morel \(2001\)](#)), and others. Later on, in [Calderero and Caselles \(2014\)](#), the axiomatic approach was extended to multiscale analyses of images defined on Riemannian manifolds. Among others, their goals were to include anisotropic scale-spaces in the common framework of multiscale analyses and to define new scale-spaces for video that take into account anisotropies and motion. Then, in [Ballester et al. \(2014\)](#), the same approach was adapted to study not scale-spaces of images, but rather multiscale similarity mea-

asures between images defined on Riemannian manifolds. Such similarity measures compare small local neighborhoods or patches which size is determined by a variable scale. Several models of multiscale analyses of patch similarity measures were studied there, and in our work we concentrate on one of them.

The key idea behind the axiomatic approach is to express desired properties of a multiscale patch similarity measure as a set of axioms. These axioms can then be turned into a family of partial differential equations and a proper similarity measure can be obtained as a solution of such a PDE. By changing the set of axioms it is possible to obtain similarity measures with different properties. Let us note, however, that some of the axioms are required in order to have a partial differential equation form of the multiscale analysis of similarity measures. As proposed in Alvarez et al. (1993), all axioms may be classified into three groups: the so called “architectural axioms”, one of a kind “comparison principle” and “morphological axioms”.

In the remainder of the chapter we first describe the set of axioms that we consider and then present theorems that link these axioms with a family of partial differential equations.

3.1 Axioms

Since a complete axiomatic classification of the different patch similarity measures is not in the scope of this work, we do not include here all possible axioms. Instead we focus only on those of them, that lead to a very practical linear model we are interested in.

In order to derive a *linear multiscale analysis of similarity measures between images on Riemannian manifolds* we work here with the product manifold $\mathcal{N} = \mathcal{M}^1 \times \mathcal{M}^2$. Let us recall from the previous chapter that the metric on this manifold is denoted by $g = g^1 \times g^2$, and that $G(\xi), G^1(\xi_1), G^2(\xi_2)$ represent the corresponding matrices in a coordinate system at given points $\xi = (\xi_1, \xi_2) \in \mathcal{N}$, $\xi_1 \in \mathcal{M}^1$, $\xi_2 \in \mathcal{M}^2$.

Let $(\kappa) := \kappa_n$ be an increasing sequence of non-negative constants. We define the following set of functions

$$\mathcal{Q}((\kappa)) := \{C \in C_b^\infty(\mathcal{N}) : \|D^\alpha C\|_\infty \leq \kappa_n \ \forall n \geq 0 \ \forall |\alpha| \leq n\}.$$

As usual, $O(f)$ denotes any expression which is bounded by $c|f|$ for some constant $c > 0$. And $o(f)$ denotes any expression such that $\frac{o(f)}{|f|} \rightarrow 0$ as

$f \rightarrow 0$. Assume that $T_t : C_b(\mathcal{N}) \rightarrow C_b(\mathcal{N})$ is a nonlinear operator for any $t \geq 0$. We shall denote $C(t, \xi) = T_t C(\xi)$, $C \in C_b(\mathcal{N})$, $\xi = (\xi_1, \xi_2) \in \mathcal{N}$, $t \geq 0$. By a slight abuse of notation, to highlight the implicit coordinate system $\psi : U_1 \times U_2 \rightarrow \mathcal{N}$, we will denote it as $T_t(C, \psi)$, which can also be written as $C(t, \psi(x, y)) = T_t(C, \psi)(x, y)$, where $x, y \in U_1 \times U_2$. Assume that we are given an a priori connection P on \mathcal{N} .

Architectural Axioms

[Recursivity] $T_0(C) = C$, $T_s(T_t C) = T_{s+t} C$, $\forall s, t \geq 0$, $\forall C \in C_b(\mathcal{N})$.

In other words, if *Recursivity* is satisfied, then $C(t_2, \xi)$ can be computed from $C(t_1, \xi)$ for any $t_2 \geq t_1$ and $T_0(C) = C$ is the identity. Notice also, that $T_t C = T_{nh} C$ when $t = nh$. This means that the effect of T_t can be achieved after n discrete iterations of T_h . The next axiom states the independence of the multiscale analysis of the choice of h .

[Infinitesimal generator]

$\frac{T_h(C, \psi) - (C, \psi)}{h} \rightarrow (\mathcal{A}(C), \psi)$ as $h \rightarrow 0+$ for any $C \in C_b^\infty(\mathcal{N})$ and any coordinate system $\psi = (\psi_1, \psi_2)$ around $\xi \in \mathcal{N}$

\mathcal{A} is the so-called infinitesimal generator for T_t . We assume that

$$T_t(R(C, \psi))(\psi^{-1}(\xi)) = R(T_t(C), \psi)(\psi^{-1}(\xi)) + o(t) = T_t(C)(\xi) + o(t) \quad (3.1)$$

as $t \rightarrow 0+$, for any $C \in C_b(\mathcal{N})$, any coordinate system $\psi = (\psi_1, \psi_2)$, and any R which are $P(\xi)$ -related rotations. We have denoted by $R(C, \psi)$ the function in the coordinate system $\bar{\psi}$ which is $P(\xi)$ -related to ψ .

By writing (3.1) in terms of the generator \mathcal{A} we have

$$\begin{aligned} R(C, \psi)(0) + t\mathcal{A}(R(C, \psi))(0) + o(t) &= R((C, \psi) + t\mathcal{A}(C, \psi))(0) + o(t) \\ &= C(\xi) + t\mathcal{A}(C, \psi)(0) + o(t). \end{aligned}$$

Using the linearity of $R(C, \psi)$, dividing by t and letting $t \rightarrow 0+$ we obtain

$$\mathcal{A}(R(C, \psi))(0) = R\mathcal{A}(C, \psi)(0) = \mathcal{A}(C, \psi)(0) \quad (3.2)$$

for any $C \in C_b(\mathcal{N})$, any coordinate system $\psi = (\psi_1, \psi_2)$, and any $P(\xi)$ -related rotations R .

Remark 3. In $T_t(R(C, \psi))$ the a priori connection is expressed in the coordinate system $\bar{\psi} = (\bar{\psi}_1, \bar{\psi}_2)$. In $R(T_t(C), \psi) = T_t(C)$ the a priori connection is expressed in the coordinate system $\psi = (\psi_1, \psi_2)$. That is, the Infinitesimal generator axiom says that both expressions are the same (intrinsic character of T_t) when the coordinate systems are $P(\xi)$ -related.

Remark 4. The Infinitesimal generator axiom contains the invariance with respect to diagonal rotations in the tangent plane of $\mathcal{M}^1 \times \mathcal{M}^2$. When $(\mathcal{M}^k, G^k) = (\mathbb{R}^N, I)$, $k \in \{1, 2\}$, it amounts to invariance with respect to Euclidean diagonal rotations in \mathbb{R}^{2N} . That is, $T_t(RC) = RT_t(C) \forall t \geq 0$, $\forall C \in C_b^\infty(\mathbb{R}^N \times \mathbb{R}^N)$, and for all $\forall R \in O(N)$ (Euclidean rotations in \mathbb{R}^N) where $RC(x, y) = C(Rx, Ry)$.

Remark 5. When both manifolds are $\mathcal{M} = \mathbb{R}^N$ with the Euclidean metric, the axioms is just:

[Infinitesimal generator] $\frac{T_h C - C}{h} \rightarrow \mathcal{A}(C)$ as $h \rightarrow 0+$. This holds for any $C \in C_b^\infty(\mathbb{R}^N \times \mathbb{R}^N)$.

In some sense the coordinate system around each point is always the same, the canonical one; they are related by the identity.

[Regularity] $\|T_t(C + h\tilde{C}) - (T_t(C) + h\tilde{C})\|_\infty \leq Mht \forall h, t \in [0, 1], \forall C, \tilde{C} \in \mathcal{Q}((\kappa))$ where the constant M depends on $\mathcal{Q}((\kappa))$.

Regularity states a natural assumption of continuity of T_t ; therefore, it is a strong justification for the existence of an infinitesimal generator for the multiscale analysis (Alvarez et al. (1993)).

[Locality] $T_t(C, \psi)(x, y) - T_t(\tilde{C}, \psi)(x, y) = o(t)$ as $t \rightarrow 0+$, $x, y \in \mathbb{R}^N$, $\forall C, \tilde{C} \in C_b(\mathcal{N})$ such that $D^\alpha C(\psi(x, y)) = D^\alpha \tilde{C}(\psi(x, y))$ for all multiindices α .

This axiom states that the value of $T_t C$ for small t , at any point ξ , is determined by the behavior of C near ξ .

Comparison Principle

[Comparison principle] $T_t C \leq T_t \tilde{C} \forall t \geq 0$ and all $C, \tilde{C} \in C_b^\infty(\mathcal{N})$ such that $C \leq \tilde{C}$.

The *comparison principle* expresses an order-preserving property. If a similarity measure \tilde{C} is everywhere bigger than another similarity measure C ,

then applying a multiscale analysis T_t does not invert this relation. Intuitively, no enhancement is made in the comparison, but just a smoothing of the original one.

Let us note at this point that *Recursivity*, *Regularity* and *Locality* axioms, together with the *Comparison principle*, are required in order to have a partial differential equation representation of a multiscale analysis.

Morphological Axioms

We are not going to consider the morphological comparison of image patches; therefore, we just mention here the *Gray level shift invariance* axiom.

[Gray level shift invariance] $T_t(0) = 0$, $T_t(C + \kappa) = T_t(C) + \kappa \forall t \geq 0$, $\forall C \in C_b^\infty(\mathcal{N})$, $\forall \kappa \in \mathbb{R}$.

This axiom states that the multiscale analysis of similarity measures is independent of the absolute zero in the comparison. In other words, patch distance between the two most similar points may well be non-zero. Although in our case of multiscale analysis of similarity measures we deal with similarity values instead of intensities, by analogy we keep for this axiom the same name introduced in Alvarez et al. (1993) for image analysis.

3.2 Similarity Measures as Solutions of PDEs

In this section we collect theorems that link together the axioms and a partial differential equation characterization of the multiscale analyses of similarity measures. We do not include here any proofs and instead refer to corresponding results from previous works.

Recall that we denote $G = (G^1, G^2)$ and $\Gamma = \Gamma^{(1)} \otimes \Gamma^{(2)}$. To simplify the notation we will not indicate explicitly the arguments for G and Γ^k .

Theorem 3.1. (i) *Let T_t be a multiscale analysis satisfying all the Architectural axioms, and the Comparison principle. Let ψ be a coordinate system around $\xi \in \mathcal{N}$. Then there exists a function $F : SM_\xi(\mathcal{N}) \times T_\xi^* \mathcal{N} \times \mathbb{R} \times \mathcal{N} \rightarrow \mathbb{R}$ increasing with respect to its first argument such that*

$$\frac{T_t(C, \psi) - (C, \psi)}{t} \rightarrow F(D^2(C \circ \psi)(0), D(C \circ \psi)(0), C(\xi), \xi, G, \Gamma^k) \text{ in } C_b(\mathcal{N})$$

as $t \rightarrow 0+$, for all $C \in C_b^\infty(\mathcal{N})$. The function F is continuous in its first three arguments.

(ii) If we assume that T_t is also gray level shift invariant, then the function F does not depend on $C(\xi)$.

This theorem states a general form of a function F , implied by the *Infinitesimal generator*. Similar theorem for multiscale analyses of images was proven in Alvarez et al. (1993) (Theorem 2) and the proof was extended to images defined on Riemannian manifolds in Calderero and Caselles (2014) (Theorem 3.2). Notice that the first argument in F is a symmetric map from $T_\xi \mathcal{N}$ to $T_\xi^* \mathcal{N}$.

Lemma 3.2. *The function F is elliptic, that is, if $A_1, A_2 : T_\xi \mathcal{N} \rightarrow T_\xi^* \mathcal{N}$ are two matrices such that A_1, A_2 are symmetric and $A_1 \leq A_2$, $p \in T_\xi^* \mathcal{N}$, $c \in \mathbb{R}$, then*

$$F(A_1, p, c, \xi, G, \Gamma^k) \leq F(A_2, p, c, \xi, G, \Gamma^k).$$

This Lemma was proven in Calderero and Caselles (2014) (Lemma 3.3).

Theorem 3.3. *Let T_t be a multiscale analysis satisfying all the Architectural axioms, the Comparison principle, and Gray level shift invariance. If $C(t, \xi) = T_t C(\xi)$, then $C(t, \xi)$ is a viscosity solution of*

$$\frac{\partial C}{\partial t} = F(D_{\mathcal{N}}^2 C, DC, \xi, G, \Gamma^k), \quad (3.3)$$

with $C(0, \xi) = C(\xi)$.

The proof of that theorem follows as in Alvarez et al. (1993) (Theorem 2), see also Guichard and Morel (2001) (Chapters 19 and 20).

Together with the previous results, the Theorem 3.3 states that a multiscale similarity measure, that satisfies the set of axioms from Section 3.1, can be obtained as a (viscosity) solution of the parabolic PDE (3.3). The PDE is given here in a very general form. In the following Chapter 4 we will go into details of the case when the operator T_t is *linear* and we will present the specific family of PDEs for that case. We will also derive practical similarity measures as approximate solutions of the corresponding PDE.

Derivation of Patch Similarity Measures

In this chapter we concentrate on a linear model, first described in [Ballester et al. \(2014\)](#). From that model we derive two patch similarity measures that are well-suited for practical applications.

We start off with a particular form of PDE, obtained from the generic one (3.3), by additionally considering the T_t operator to be linear. Then we make an assumption about the manifolds that is reasonable in the context of image and video processing. Namely, we consider $\mathcal{M}^i = \mathbb{R}^N$ endowed with a general metric. Then we study a simple case, when the metrics on manifolds are constant all over the image domains. And finally, we apply the WKB approximation method to derive the similarity measure that takes into account spatially varying metrics.

4.1 Linearity Assumption

From now on we consider the operator T_t to be linear, that is

$$T_t(aC_1 + bC_2) = aT_t(C_1) + bT_t(C_2), \quad \forall a, b \in \mathbb{R}, \quad \forall C_1, C_2 \in C_b(\mathcal{N}).$$

Notice that this can be seen as an additional *Linearity* axiom which should be satisfied by the multiscale analysis of similarity measures.

Theorem 4.1. *Let T_t be a multiscale analysis of similarity functions satisfying all Architectural axioms, the Comparison principle, and Gray level*

shift invariance axiom. Assume also that T_t is linear. Then

$$\frac{\partial C}{\partial t} = F(D_{\mathcal{N}}^2 C, \xi, G),$$

where

$$\begin{aligned} F(X, \xi, G) = & c_{11}(\xi) \text{Tr}((G^1)^{-1}(\xi_1) X_{11}) + 2c_{12}(\xi, G) \text{Tr}(\bar{D}_{12} I^1(\xi_1)^{-1} X_{12}) \\ & + c_{22}(\xi) \text{Tr}((G^2)^{-1}(\xi_2) X_{22}), \end{aligned}$$

and \bar{D}_{12} is an isometry from $(T_{\xi_1} \mathcal{M}^1, G^1(\xi_1)) \rightarrow (T_{\xi_2} \mathcal{M}^2, G^2(\xi_2))$. The ellipticity of F implies that $c_{11}, c_{22} \geq 0$.

Moreover,

$$2c_{12}(\xi, G) \bar{D}_{12} I^1(\xi_1)^{-1} = B^2(\xi_2) D' B^1(\xi_1)^t, \quad (4.1)$$

and the dependence of $c_{12}(\xi, G) \bar{D}_{12} I^1(\xi_1)^{-1}$ on G is only in isometries $B^2(\xi_2)$ and $B^1(\xi_1)^t$. Here D' is a matrix that only depends on ξ (see Ballester et al. (2014)). We could also write the second term as

$$2c_{21}(\xi, G) \text{Tr}(\bar{D}_{21} I^2(\xi_2)^{-1} X_{21}).$$

Notice that for $X = D_{\mathcal{N}}^2 C$ we have $X_{ii} = D_{\mathcal{M}_i}^2 C$ and that the operators $c_{ii}(\xi) \text{Tr}((G^i)^{-1}(\xi_i) X_{ii})$ are multiples of the Laplace-Beltrami operator. Notice also that there are no first order terms in these operators. They cannot couple with vectors, and thus we have the required invariance induced by the rotations in tangent planes.

Theorem 4.1 was originally proved in Ballester et al. (2014). Since it is of great importance for our work, we include the proof of it in Appendix A.

4.2 The Case of $(\mathcal{M}^k, g^k) = (\mathbb{R}^N, g^k)$

To fix ideas we consider $\mathcal{M}^1 = \mathcal{M}^2 = \mathcal{M} = \mathbb{R}^N$ and $g_{ij}^k(x)$ be general metrics in \mathbb{R}^N , $k = 1, 2$, $x \in \mathcal{M}^k$. We know that $e_i = G^k(x)^{-\frac{1}{2}} f_i$ is a orthonormal basis of $(T_x \mathcal{M}^k, g^k(x))$, if f_i is a Euclidean orthonormal basis. Indeed, for $i, j \in \{1, \dots, N\}$, we have

$$\begin{aligned} \langle e_i, e_j \rangle_{\mathcal{M}^k} &= \langle G^k(x) e_i, e_j \rangle_{\mathbb{R}^N, \text{Eucl}} = \langle G^k(x)^{\frac{1}{2}} e_i, G^k(x)^{\frac{1}{2}} e_j \rangle_{\mathbb{R}^N, \text{Eucl}} \\ &= \langle f_i, f_j \rangle_{\mathbb{R}^N, \text{Eucl}} = \delta_{ij}. \end{aligned}$$

Let $I^k(x) : (\mathbb{R}^N, g^k(x)) \rightarrow (\mathbb{R}^N, (g^k)^{-1}(x))$ be given by $I^k(x)e_i = e_i^*$. Then

$$I^k(x) = G^k(x).$$

If $B^k(x)$ satisfies $B^k(x)I^k(x)^{-1}B^k(x)^t = G^k(x)^{-1}$, then we can take $B^k(x) = I$.

We can define $P(x, y)(v) = G^2(y)^{-\frac{1}{2}}G^1(x)^{\frac{1}{2}}v$, $v \in \mathbb{R}^N$, as the a priori connection of x and y . Then $|P(x, y)v|_{g^2} = |v|_{g^1}$ for all $(x, y) \in \mathbb{R}^{2N}$. Recall that $\bar{D}_{1,2} : (\mathbb{R}^N, g^1(x)) \rightarrow (\mathbb{R}^N, g^2(y))$ is an isometry, in this case it is given by $\bar{D}_{1,2} = G^2(y)^{-\frac{1}{2}}G^1(x)^{\frac{1}{2}}$. Then (4.1) is

$$2c_{12}(x, y)\bar{D}_{1,2}I^1(x)^{-1} = 2c_{12}(x, y)G^2(y)^{-\frac{1}{2}}G^1(x)^{-\frac{1}{2}}.$$

The PDE obtained is

$$\frac{\partial C}{\partial t} = a(x, y)\Delta_{\mathcal{M}x}C + 2c_{12}(x, y)\text{Tr}(G^2(y)^{-\frac{1}{2}}G^1(x)^{-\frac{1}{2}}D_{xy}C) + c(x, y)\Delta_{\mathcal{M}y}C, \quad (4.2)$$

where

$$\Delta_{\mathcal{M}x}C = \text{Tr}(G^1(x)^{-1}(D_{xx}C(x) - \Gamma^{(1)}(D_xC)(x))).$$

Similarly for the operator $\Delta_{\mathcal{M}y}$.

Remark 6. Note that (first by transposition, then by reordering), we have

$$\begin{aligned} \text{Tr}(G^2(y)^{-\frac{1}{2}}G^1(x)^{-\frac{1}{2}}D_{xy}C) &= \text{Tr}(D_{yx}CG^1(x)^{-\frac{1}{2}}G^2(y)^{-\frac{1}{2}}) = \\ &= \text{Tr}(G^1(x)^{-\frac{1}{2}}G^2(y)^{-\frac{1}{2}}D_{yx}C), \end{aligned}$$

which is a symmetric expression in (x, y) . If T_t is symmetric in (x, y) , then c_{12} is also symmetric.

In the symmetric case, the matrix associated with the operator (4.2) is

$$\begin{bmatrix} a(x, y)G^1(x)^{-1} & c_{12}(x, y)G^2(y)^{-\frac{1}{2}}G^1(x)^{-\frac{1}{2}} \\ c_{12}(x, y)G^1(x)^{-\frac{1}{2}}G^2(y)^{-\frac{1}{2}} & c(x, y)G^2(y)^{-1} \end{bmatrix}.$$

It is positive semidefinite if and only if $a, c \geq 0$ and $ac - c_{12}^2 \geq 0$.

4.3 Similarity Measure with Constant Metrics

In this section we consider a simple case among the models described in Theorem 4.1 which corresponds to the situation when the metrics are constant in both images and we demonstrate an example of patch similarity measure solving that equation.

Let u and v be two given images. Let A and B be two $N \times N$ matrices. Then a simple PDE is given by

$$\frac{\partial C}{\partial t} = \text{Tr}(AA^t D_x^2 C) + 2\text{Tr}(AB^t D_{xy} C) + \text{Tr}(BB^t D_y^2 C). \quad (4.3)$$

It can be shown, that (4.3) is satisfied by the following multiscale similarity measure

$$C(t, x, y) = \int_{\mathbb{R}^N} g_t(h) C(0, x + Ah, y + Bh) dh, \quad (4.4)$$

where g_t is the Gaussian of scale t , and

$$C(0, x, y) = (u(x) - v(y))^2.$$

Indeed,

$$\begin{aligned} \frac{\partial C}{\partial t} &= \int_{\mathbb{R}^N} \frac{\partial}{\partial t} (g_t(h)) C(0, x + Ah, y + Bh) dh \\ &= \int_{\mathbb{R}^N} \Delta_h g_t(h) C(0, x + Ah, y + Bh) dh \\ &= \int_{\mathbb{R}^N} g_t(h) \Delta_h C(0, x + Ah, y + Bh) dh \\ &= \int_{\mathbb{R}^N} g_t(h) \Delta_h (u(x + Ah) - v(y + Bh))^2 dh. \end{aligned}$$

Note that

$$\begin{aligned} \Delta_h (u(x + Ah) - v(y + Bh))^2 &= 2 \|\nabla_h (u(x + Ah) - v(y + Bh))\|^2 \\ + 2(u(x + Ah) - v(y + Bh)) \Delta_h (u(x + Ah) - v(y + Bh)) &=: I + II, \end{aligned}$$

where we have denoted both previous terms as I and II in order to compute them separately. For notation simplicity we omit the arguments of u and

v . Then

$$\begin{aligned}
II &= 2(u-v)\Delta_h(u-v) = 2(u-v) (\text{Tr}(AA^t D_x^2 u) - \text{Tr}(BB^t D_y^2 v)) \\
&= 2(u-v) (\text{Tr}(AA^t D_x^2(u-v)) + \text{Tr}(BB^t D_y^2(u-v))) \\
&= \text{Tr}(AA^t D_x^2(u-v)^2) - 2\text{Tr}(AA^t \nabla_x(u-v) \otimes \nabla_x(u-v)) \\
&+ \text{Tr}(BB^t D_y^2(u-v)^2) - 2\text{Tr}(BB^t \nabla_y(u-v) \otimes \nabla_y(u-v)) \\
&= \text{Tr}(AA^t D_x^2(u-v)^2) - 2\|A^t \nabla_x u\|^2 + \text{Tr}(BB^t D_y^2(u-v)^2) \\
&- 2\|B^t \nabla_y v\|^2
\end{aligned}$$

and

$$\begin{aligned}
I &= 2\|\nabla_h(u-v)\|^2 = 2\|A^t \nabla_x u - B^t \nabla_y v\|^2 \\
&= 2\|A^t \nabla_x u\|^2 + 2\|B^t \nabla_y v\|^2 - 4\langle A^t \nabla_x u, B^t \nabla_y v \rangle.
\end{aligned}$$

Then

$$\begin{aligned}
I + II &= \text{Tr}(AA^t D_x^2(u-v)^2) + \text{Tr}(BB^t D_y^2(u-v)^2) - 4\langle A^t \nabla_x u, B^t \nabla_y v \rangle \\
&= \text{Tr}(AA^t D_x^2(u-v)^2) + \text{Tr}(BB^t D_y^2(u-v)^2) + 2\text{Tr}(AB^t D_{xy}(u-v)^2).
\end{aligned}$$

Thus,

$$\frac{\partial C}{\partial t} = \text{Tr}(AA^t D_x^2 C) + \text{Tr}(BB^t D_y^2 C) + 2\text{Tr}(AB^t D_{xy} C).$$

4.4 Approximate Solution for Varying Metrics

Let us first analyse the example of similarity measure with constant metrics from the previous section. We start by writing the operator in (4.3) as

$$\begin{aligned}
\frac{\partial C}{\partial t} &= \text{Tr}(AA^t D_x^2 C) + 2\text{Tr}(AB^t D_{xy} C) + \text{Tr}(BB^t D_y^2 C). \\
&= \text{Tr} \left(\begin{bmatrix} AA^t & AB^t \\ BA^t & BB^t \end{bmatrix} \begin{bmatrix} D_{\mathcal{N},xx} C & D_{\mathcal{N},xy} C \\ D_{\mathcal{N},xy} C & D_{\mathcal{N},yy} C \end{bmatrix} \right).
\end{aligned}$$

Note that

$$\begin{bmatrix} AA^t & AB^t \\ BA^t & BB^t \end{bmatrix} = \begin{bmatrix} A & 0 \\ B & 0 \end{bmatrix} \begin{bmatrix} A^t & B^t \\ 0 & 0 \end{bmatrix} =: \Sigma \Sigma^t.$$

Note that neither Σ nor $\Sigma \Sigma^t$ are invertible operators and we would need to regularize them, for instance, by introducing a perturbation ϵI in the $(2, 2)$ entry of Σ .

Remark 7. Note that the PDE above can be related to the stochastic ODE:

$$\begin{aligned} dX_t &= AdW_t \\ dY_t &= BdW_t \end{aligned} \quad (4.5)$$

where the Brownian motion dW_t is common in both equations.

The more general linear case derived from (4.2) is

$$\frac{\partial C}{\partial t} = \Delta_{\mathcal{M}_x} C + 2\text{Tr}(G^2(y)^{-\frac{1}{2}} G^1(x)^{-\frac{1}{2}} D_{xy} C) + \Delta_{\mathcal{M}_y} C, \quad (4.6)$$

where

$$\begin{aligned} \Delta_{\mathcal{M}_x} C &= \text{Tr}(G^1(x)^{-1} (D_{xx} C - \Gamma^{(1)}(D_x C)(x, y))), \\ \Delta_{\mathcal{M}_y} C &= \text{Tr}(G^2(y)^{-1} (D_{yy} C - \Gamma^{(2)}(D_y C)(x, y))). \end{aligned}$$

It involves spatially varying metrics and can be subsumed under the previous notation by taking

$$A = G^1(x)^{-\frac{1}{2}}, \quad B = G^2(y)^{-\frac{1}{2}}.$$

Notice that we do not necessarily assume A and B to be constant, even though we write here A and B instead of $A(x)$ and $B(y)$. _____

In any case, Σ to be invertible, let us write (4.3) as

$$\frac{\partial C}{\partial t} = \text{Tr}(\Sigma \Sigma^t D_{\mathcal{N}}^2 C) =: \text{Tr}_{(\Sigma \Sigma^t)^{-1}}(D_{\mathcal{N}}^2 C) \quad (4.7)$$

where $\text{Tr}_{(\Sigma \Sigma^t)^{-1}}(D_{\mathcal{N}}^2 C)$ is a notation for the Laplace-Beltrami operator (eventually degenerated), that is, the trace of the Hessian with respect to the (eventually degenerated) metric $g := (\Sigma \Sigma^t)^{-1}$. In this case, the result in Varadhan (1967) could give the approximate formula we are looking for. However, we will use another approach.

We would like to obtain an approximation formula of the type

$$C(t + \epsilon, p) = \int K(\epsilon, p, p') C(t, p') dp'$$

where $p = (x, y)$, $p' = (x', y')$.

We proceed using the so called WKB approximation as in Sochen et al. (2001) (Sections 3 and Appendix). Let us recall that WKB theory, named after Wentzel, Kramers and Brillouin, is well known in quantum mechanics

and is used to find approximate solutions to linear partial differential equations with spatially varying coefficients. Following the WKB method, we assume the kernel K to be of the form

$$K(t, p, p') = \frac{H(t, p, p')}{\sqrt{t}} e^{-\Psi(p, p')/t}.$$

We are interested in the behavior of the kernel for t small. Without loss of generality, we can assume that $H(t, p, p') = H_0$, a constant. Indeed, as in [Sochen et al. \(2001\)](#) (Appendix) one can check that for t small the leading term corresponds to $H = H_0$ constant. The function Ψ does not depend on t and is positive. The validity of this approximation procedure can be found, for example, in [Cohen et al. \(1972\)](#).

To simplify the notation, let us forget the arguments of the above functions. Since the equation is linear we may assume that it is satisfied by the kernel K . Then, introducing K into the equation (4.7), some straightforward computations produce an equality with several terms. For short times only the most singular part is dominant. Therefore, by considering the leading terms of order $\frac{1}{t^{5/2}}$ of both sides of the equation, which are the most divergent ones as $t \rightarrow 0+$, we have that the term on the left hand side (obtained from $\frac{\partial K}{\partial t}$) is

$$\frac{H_0}{\sqrt{t}} e^{-\Psi/t} \frac{\Psi}{t^2}$$

and the term obtained from the right hand side is

$$\frac{H_0}{\sqrt{t}} e^{-\Psi/t} \frac{1}{t^2} \langle \Sigma \Sigma^t \nabla_p \Psi, \nabla_p \Psi \rangle.$$

The equality of both terms gives the PDE:

$$\langle \Sigma \Sigma^t \nabla_p \Psi, \nabla_p \Psi \rangle = \Psi,$$

that is,

$$\|\Sigma^t \nabla_p \Psi\|^2 = \Psi. \quad (4.8)$$

If $\Phi = 2\sqrt{\Psi}$, (4.8) becomes

$$\|\Sigma^t \nabla_p \Phi\|^2 = 1. \quad (4.9)$$

From $\Sigma^t = \begin{bmatrix} A^t & B^t \\ 0 & 0 \end{bmatrix}$, we can write the above equation (4.9) as

$$\|A^t D_x \Phi + B^t D_y \Phi\|^2 = 1. \quad (4.10)$$

Let us denote $(P, Q) = (D_x \Phi, D_y \Phi)$, and $H(P, Q) = \|A^t P + B^t Q\|^2$. Writing the Hamilton-Jacobi equation (4.10) in terms of H , the solution of

$$H(P, Q) = 1$$

is given in terms of the Lagrangian

$$L(\bar{P}, \bar{Q}) = \sup_{(P, Q)} \{ \langle (\bar{P}, \bar{Q}), (P, Q) \rangle - H(P, Q) \}.$$

Note that, with

$$\begin{aligned} R &= A^t P + B^t Q, & S &= A^t P - B^t Q, \\ \bar{R} &= A^{-1} \bar{P} + B^{-1} \bar{Q}, & \bar{S} &= A^{-1} \bar{P} - B^{-1} \bar{Q}, \end{aligned}$$

we may write

$$L(\bar{P}, \bar{Q}) = \sup_{(R, S)} \left\{ \frac{1}{2} \langle (\bar{R}, \bar{S}), (R, S) \rangle - \|R\|^2 \right\}.$$

Then,

$$\text{if } \bar{S} \neq 0, \text{ then } L(\bar{P}, \bar{Q}) = +\infty,$$

$$\text{if } \bar{S} = 0, \text{ then } L(\bar{P}, \bar{Q}) = \frac{1}{16} \|\bar{R}\|^2 = \frac{1}{16} \|A^{-1} \bar{P} + B^{-1} \bar{Q}\|^2.$$

Recall that the solution of the Hamilton-Jacobi equation (4.10) is given by

$$\Phi(t, p, p') = \frac{1}{16} \inf_{\mathcal{C}} \int_p^{p'} \|A^{-1} \dot{\gamma}(s) + B^{-1} \dot{\tilde{\gamma}}(s)\|^2 ds,$$

where $p = (x, y)$, $p' = (x', y')$, \mathcal{C} is the set of curves $(\gamma, \tilde{\gamma})$ such that γ is a curve joining x to x' and $\tilde{\gamma}$ a curve joining y to y' , and

$$A^{-1} \dot{\gamma}(s) = B^{-1} \dot{\tilde{\gamma}}(s),$$

that is,

$$\dot{\gamma}(s) = AB^{-1} \dot{\tilde{\gamma}}(s). \quad (4.11)$$

The observations above are valid for any metrics being constant or not. Let us analyze the last formula when A and B are constant matrices. By integrating (4.11) we get

$$x' - x = AB^{-1}(y' - y),$$

that is,

$$B^{-1}(y' - y) = A^{-1}(x' - x) := h.$$

Thus, we may write

$$x' - x = Ah, \quad y' - y = Bh.$$

Moreover,

$$\Phi(t, p, p') = \frac{1}{4} \inf_{\gamma} \int_x^{x'} \|A^{-1}\dot{\gamma}(s)\|^2 ds \left(= \frac{1}{4} \inf_{\tilde{\gamma}} \int_y^{y'} \|B^{-1}\dot{\tilde{\gamma}}(s)\|^2 ds \right). \quad (4.12)$$

Let us solve this equation explicitly in the case of constant matrices A, B . The solution is given by γ being a straight line joining its two endpoints. Indeed, writing $\alpha(s) := A^{-1}\gamma(s)$, then

$$\Phi(t, p, p') = \frac{1}{4} \inf_{\alpha} \int_{A^{-1}x}^{A^{-1}x'} \|\dot{\alpha}(s)\|^2 ds. \quad (4.13)$$

The solution is given by

$$\Phi(t, p, p') = \frac{1}{4} \|A^{-1}x - A^{-1}x'\|^2 = \frac{1}{4} \|h\|^2. \quad (4.14)$$

Recall $\Phi = 2\sqrt{\Psi}$. Then, for $t > 0$ small enough, we have the approximation

$$C(t, p) = \frac{H_0}{\sqrt{t}} \int e^{-\Psi(t, p, p')/t} C(0, x + Ah, y + Bh) dh$$

for some constant H_0 . Then, after adjusting constants,

$$C(t, p) = \int g_t(h) C(0, x + Ah, y + Bh) dh,$$

where $g_t(h)$ denotes the Gaussian of variance t , and we recover the formula given in Section 4.3.

In the general case, where A and B are not constant matrices but $A(x) = G^1(x)^{-\frac{1}{2}}$, $B(y) = G^2(y)^{-\frac{1}{2}}$, we have

$$\begin{aligned} \Phi(t, p, p') &= \frac{1}{4} \inf_{\gamma} \int_x^{x'} \|G^1(\gamma(s))^{\frac{1}{2}} \dot{\gamma}(s)\|^2 ds = \\ &= \frac{1}{4} \inf_{\tilde{\gamma}} \int_y^{y'} \|G^2(\tilde{\gamma}(s))^{\frac{1}{2}} \dot{\tilde{\gamma}}(s)\|^2 ds. \end{aligned} \quad (4.15)$$

If we interpret (4.15) intuitively as a geodesic distance $d(x, x')$ in the manifold \mathcal{M}^1 , and set $x' = x + A(x)h$, then

$$C(t, p) = \frac{H_0}{\sqrt{t}} \int e^{-d(x, x+A(x)h)^2/t} C(0, x + A(x)h, y + B(y)h) dh.$$

After adjusting constants and substituting back the metrics we finally obtain

$$C(t, p) = \int e^{-d(x, x+G^1(x)^{-\frac{1}{2}}h)^2/t} C(0, x + G^1(x)^{-\frac{1}{2}}h, y + G^2(y)^{-\frac{1}{2}}h) dh. \quad (4.16)$$

The geodesic distance can be approximated as in the bilateral filter in Tomasi and Manduchi (1998) by

$$\begin{aligned} d(x, x + G^1(x)^{-\frac{1}{2}}h)^2 \\ = \kappa_{spatial} \|G^1(x)^{-\frac{1}{2}}h\|^2 + \kappa_{color} |u(x) - u(x + G^1(x)^{-\frac{1}{2}}h)|^2, \end{aligned}$$

where $\kappa_{spatial} > 0$, $\kappa_{color} > 0$, and u is the image on \mathcal{M}^1 .

On the other hand, by a drastic approximation, writing $A = G^1(x)^{-\frac{1}{2}}$, $B = G^2(y)^{-\frac{1}{2}}$, $h = A^{-t}x' - A^{-t}x$, we can also obtain the formula that coincides with the one for constant metrics A and B

$$C(t, p) = \int g_t(h) C(0, x + G^1(x)^{-\frac{1}{2}}h, y + G^2(y)^{-\frac{1}{2}}h) dh. \quad (4.17)$$

Recall that in both (4.16) and (4.17) we consider $p = (x, y)$ and

$$C(0, x, y) = (u(x) - v(y))^2.$$

Formulas (4.16) and (4.17) represent the patch similarity measures that are well suited for practical applications. The difference between them is in the weighting function. Approximated geodesic weights are used in the first case, while simpler Gaussian weights are used in the second case. In the following chapter it will be shown that these similarity measures can in fact be affine invariant. Both of them will be discussed and tested in Sections 7.1 and 7.2. Then in Part II of this dissertation the computationally less expensive (4.17) will be exploited for image inpainting and image denoising applications.

Structure Tensor as a Metric

In the previous chapter we have derived two patch similarity measures, namely (4.16) and (4.17), with some Riemannian metrics incorporated in them. It is well known, that the so called structure tensor (also known as second-moment matrix) can be seen as a metric in the image domain (e.g. Weickert (1998, 1999); Kimmel et al. (2000); Brox et al. (2006b,a)). It is used in image processing and computer vision fields in applications ranging from nonlinear filtering to motion analysis.

In this chapter we address the construction of *affine covariant structure tensors* and discuss their properties. As will be shown, the proposed iterative scheme gives structure tensors that transform properly by an affine transformation; therefore, we additionally title them as “affine covariant”. The a priori connection, built from affine covariant structure tensors, allows us to compare vicinities of two given points regardless of any affine transformation between them. In order to stress this property, we will denote the proposed patch similarity measures built with affine covariant structure tensors as $\mathcal{D}^a(t, x, y)$ and name them *affine invariant patch similarity measures*. That is, we explicitly say that the similarity value between two patches does not change under affine transformations.

The computation of affine covariant structure tensors is closely related to the problem of estimating affine covariant regions. The latter was addressed before in the object recognition literature. In Mikolajczyk and Schmid (2004), the authors compute affine covariant regions on a set of points that are robust to scale changes. Given two matching points, the affine transformation can then be estimated up to a rotation. In Matas et al. (2004), the authors build up affine covariant domains referred to as maximally stable extremal

regions (MSER). Mainly, MSER are defined as the most contrasted connected components of upper and lower level sets of the image. Even though these approaches give very good results in the object recognition context, they do not provide a dense set of regions with a guarantee of their affine covariance. In the following section we propose a way of calculating affine covariant regions directly from the dense field of affine covariant structure tensors.

We begin this chapter with a general definition of the affine covariant structure tensors and neighborhoods in Section 5.1. We also propose an iterative construction scheme that defines particular affine covariant structure tensors and neighborhoods. We show that the proposed structure tensors are well-suited to be used as metrics in the affine invariant similarity measures. In Section 5.2 we put together the affine covariant structure tensors and the concept of a priori connection. Then in Section 5.3 we briefly discuss limitations of the proposed structure tensors computed in real discrete images. In Section 5.4 we empirically study the convergence and dependency on initial conditions of the iterative construction scheme. Finally, in Section 5.5 we compare in terms of performance the proposed affine covariant neighborhoods with a related approach from the literature.

5.1 Affine Covariant Structure Tensors

In this section we describe the construction of structure tensors and their corresponding neighborhoods which are affine covariant.

Let u be a given image, $u : \mathbb{R}^N \rightarrow \mathbb{R}$. Let $GL(N)$ be the set of invertible matrices in \mathbb{R}^N . Let $A \in GL(N)$ be an affine transformation. We denote by $u_A(x) := u(Ax)$ a version of an image u , transformed by an affinity A . Notice that it is equivalent to say that A transforms the basis of u to the basis of u_A .

We begin with some examples of the kind of tensors we are interested in. Assume that we have a metric g on \mathbb{R}^N . The map A induces another metric g_A in \mathbb{R}^N such that $G_A(x) = A^t G(Ax) A$. As always $G(x)$ denotes the symmetric matrix $[g_{ij}]$ known as *metric tensor*. The law of transformation from G to G_A is implied by the properties of metric tensor. Hessian, which as well can be seen as tensor, is the next relevant example. It can be shown that the same law of transformation holds for the Hessian of an image u and the Hessian of its transformed version u_A . Indeed, $D^2 u_A(x) = A^t D^2 u(Ax) A$. Obviously it is also the case when we consider the Hessian

of u , defined on the Riemannian manifold $(\mathcal{M}^1, g^1) := (\mathbb{R}^N, g)$, and the Hessian of u_A , defined on the Riemannian manifold $(\mathcal{M}^2, g^2) := (\mathbb{R}^N, g_A)$. Indeed, $D_{\mathcal{M}^2}^2 u_A(x) = A^t D_{\mathcal{M}^1}^2 u(Ax)A$. The following definition generalizes this property for a $(1, 1)$ tensor field computed from an image.

Definition 5.1. Let H_u be a $(1, 1)$ tensor defined on \mathbb{R}^N such that, for each $x \in \mathbb{R}^N$, it is represented by a $N \times N$ matrix $H_u(x)$ mapping a vector in \mathbb{R}^N to another vector in \mathbb{R}^N . We say that H_u is an affine covariant tensor if it satisfies

$$H_{u_A}(x) = A^t H_u(Ax)A, \quad (5.1)$$

where $u_A(x) := u(Ax)$ for $A \in GL(N)$.

Let us highlight a slight clash of terminology that we commit here. In general the term “covariant tensor” is used to denote any tensor of type $(0, n)$, that is, any tensor that has n covariant indices and no contravariant indices. On the other hand, in the context of Definition 5.1 we say “affine covariant” to emphasize that such tensor, computed from an image u , transforms in accordance with an affinity.

Another interesting example is $F(u) = Du \otimes Du$, where \otimes denotes the tensor product. Then,

$$\begin{aligned} F(u_A)(x) &= Du_A(x) \otimes Du_A(x) = A^t Du(Ax) \otimes A^t Du(Ax) \\ &= A^t Du(Ax) \otimes Du(Ax)A = A^t F(u)(Ax)A. \end{aligned} \quad (5.2)$$

Thus, $F(u)$ is an affine covariant tensor.

This law of transformation is well adapted to define neighborhoods that transform properly with respect to affine transformations.

Lemma 5.2. Let H_u be an affine covariant tensor. Let

$$B_{H_u}(x, r) = \{y : \langle H_u(x)(y - x), (y - x) \rangle \leq r^2\} \quad x \in \mathbb{R}^N, r > 0. \quad (5.3)$$

Then, $B_{H_{u_A}}(x, r) = A^{-1}B_{H_u}(Ax, r)$.

We say that $B_{H_{u_A}}(x, r)$ is an *affine covariant neighborhood*.

Proof. Using (5.1) we can write

$$\begin{aligned} B_{H_{u_A}}(x, r) &= \{y : \langle H_{u_A}(x)(y - x), (y - x) \rangle \leq r^2\} \\ &= \{y : \langle A^t H_u(Ax)A(y - x), (y - x) \rangle \leq r^2\} \end{aligned}$$

Let $\bar{x} = Ax$, $\bar{y} = Ay$. Then

$$B_{H_{u_A}}(x, r) = \{A^{-1}\bar{y} : \langle H_u(\bar{x})(\bar{y} - \bar{x}), (\bar{y} - \bar{x}) \rangle \leq r^2\} = A^{-1}B_{H_u}(Ax, r).$$

□

In particular, if we define

$$B_u(x, r) = \{y : |Du(x)(y - x)| \leq r\}, \quad (5.4)$$

then

$$\begin{aligned} B_{u_A}(x, r) &= \{y : |Du_A(x)(y - x)| \leq r\} \\ &= A^{-1}\{y : y \in B_u(Ax, r)\} = A^{-1}B_u(Ax, r), \end{aligned}$$

that is, $B_{u_A}(x, r)$ is an affine covariant neighborhood as well.

Iterative Construction Scheme

At this point we have all the ingredients we need to describe the scheme for construction of affine covariant structure tensors and affine covariant neighborhoods.

Let $B_u(x, r)$ be an affine covariant neighborhood in image u . For example, $B_u(x, r)$ can be computed using (5.3) or (5.4).

Let

$$T(u)(x) = \int_{B_u(x, r)} Du(y) \otimes Du(y) dy. \quad (5.5)$$

Then

$$\begin{aligned} T(u_A)(x) &= \int_{B_{u_A}(x, r)} Du_A(y) \otimes Du_A(y) dy \\ &= \int_{A^{-1}B_u(Ax, r)} A^t Du(Ay) \otimes Du(Ay) A dy \end{aligned} \quad (5.6)$$

and by writing $\bar{y} = Ay$, $y \in A^{-1}B_u(Ax, r)$ we get

$$T(u_A)(x) = A^t \int_{B_u(Ax, r)} Du(\bar{y}) \otimes Du(\bar{y}) |\det A|^{-1} d\bar{y} A. \quad (5.7)$$

Notice that $T(u_A)(x) = |\det A|^{-1} A^t T(u)(Ax) A$; therefore, $T(u_A)$ is an affine covariant tensor with a weight expressed by $|\det A|^{-1}$. We still refer to it as an affine covariant tensor density of exponent -1 .

Although the following results hold for the general case of \mathbb{R}^N , we will focus on the case of regular images where $B_u(x, r) \subset \mathbb{R}^2$. To cancel the factor $|\det A|^{-1}$, we observe that $\text{Area}(B_{u_A}(x, r)) = |\det A|^{-1} \text{Area}(B_u(Ax, r))$. Therefore, if we normalize $T(u)(x)$ and define

$$NT(u)(x) = \frac{\int_{B_u(x, r)} Du(y) \otimes Du(y) dy}{\text{Area}(B_u(x, r))}, \quad (5.8)$$

we have $NT(u_A)(x) = A^t NT(u)(Ax)A$. In other words, $NT(u)$ is an affine covariant tensor (or an affine covariant tensor density of exponent 0), computed on an affine covariant neighborhood.

Lemma 5.3. *Let H^1 be an affine covariant tensor density of exponent p ($p = 0, -1$) and let H^2 be an affine covariant tensor. Let H_A^i be the tensor after the affine transformation A . Let $B_{H^1}(x, r)$ be an affine covariant neighborhood, computed from H^1 . Let*

$$T(H^1, H^2)(x) = \int_{B_{H^1}(x, r)} H^2(y) dy. \quad (5.9)$$

Then

$$T(H_A^1, H_A^2)(x) = |\det A|^p A^t T(H^1, H^2)(Ax)A. \quad (5.10)$$

That is, $T(H^1, H^2)$ is an affine covariant tensor density of exponent p .

We have taken $p = 0, -1$ because we wanted to cover our examples. Other exponents could be taken as well.

Lemma 5.3 permits to iterate the above construction (5.8) and redefine for $k \geq 1$

$$NT^{(k)}(u)(x) = \frac{\int_{B_{NT^{(k-1)}(u)}(x, r)} Du(y) \otimes Du(y) dy}{\text{Area}(B_{NT^{(k-1)}(u)}(x, r))} \quad (5.11)$$

where k is the index of iteration, and

$$B_{NT^{(k)}(u)}(x, r) = \{y : \langle NT^{(k)}(u)(x)(y - x), (y - x) \rangle \leq r^2\} \quad (5.12)$$

for $k \geq 1$,

$$B_{NT^0(u)}(x, r) = \{y : |Du(x)(y - x)| \leq r\} \quad (5.13)$$

for $k = 0$.

Equations (5.11), (5.12) and (5.13) constitute an iterative scheme for calculation of affine covariant structure tensors and neighborhoods.

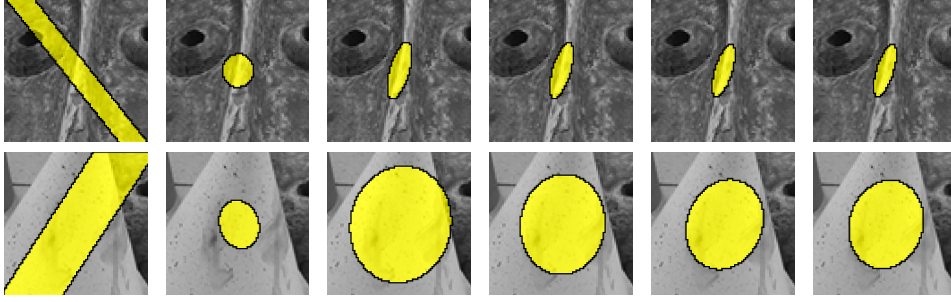


Figure 5.1: Evolution of affine covariant neighborhoods over iterations of the construction scheme (k from 0 to 5).

Notice that the initial neighborhood (5.13) takes into account non-local information due to its infinite band support. Moreover, its shape depends only on the gradient at a single point and thus may be subjected to noise. The iterative process decreases this dependency of the structure tensor on the initial neighborhood $B_{NT^0(u)}(x, r)$. Figure 5.1 illustrates evolution of affine covariant neighborhoods over iterations. In Section 5.4 we empirically study the convergence of the iterative scheme and the above mentioned dependency on the initial iteration.

To simplify the notation we will usually denote by $T_u(x)$ the affine covariant structure tensor $NT^{(k)}(u)(x)$ for a fixed number of iterations k and a given value of r . We say that T_u is the affine covariant structure tensor field associated with u . Similarly we will use the notation $B_{T_u}(x)$ to refer to the affine covariant neighborhood $B_{NT^{(k)}(u)}(x, r)$.

Let us note that r is a free parameter. It controls the size of the affine covariant neighborhood at a given point. On the other hand, the size of the neighborhood is also affected by the texture in the vicinity of that point. Some examples of the affine covariant neighborhoods, computed using the same value of r , are shown in Figure 5.2.

Remark 8. Another potentially interesting approach to the initial neighborhood calculation would be

$$\bar{B}_u(x, r) = \{y : |Du(y)(y - x)| \leq r\}. \quad (5.14)$$

Notice that in contrast to (5.13), the gradient here is taken at the point y ; therefore, such neighborhood does not depend solely on the point x . Since

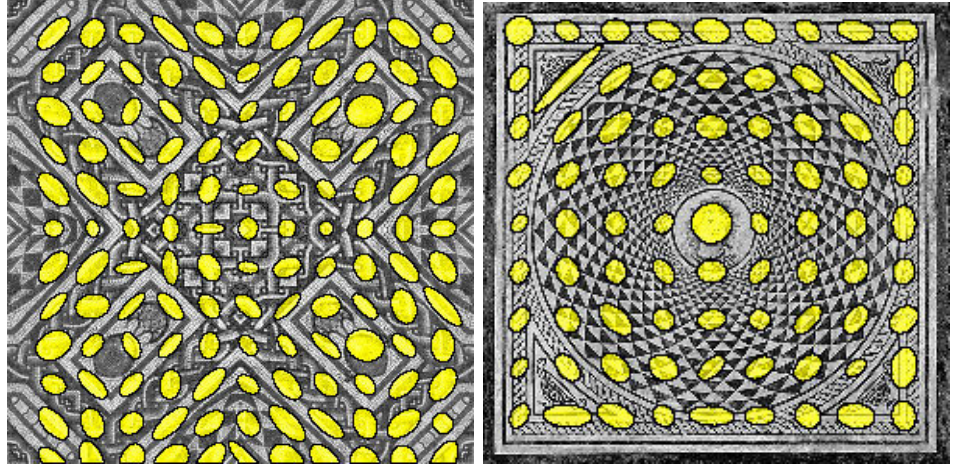


Figure 5.2: Affine covariant neighborhoods computed every 25 pixels using the same values of r .

$\bar{B}_{u_A}(x, r) = A^{-1}\bar{B}_u(Ax, r)$, we get an affine covariant neighborhood. This approach is yet to be studied in the future work.

5.2 A Priori Connection from Structure Tensors

In this section we link together the affine covariant structure tensors and the a priori connection, defined earlier in Section 2.2. Even though the a priori connection appears only implicitly in the patch similarity measures (4.16) and (4.17), the concept is still of great importance for both theoretical and practical aspects of this work. To give a better geometrical intuition for the a priori connection, we study in detail its construction from the affine covariant structure tensors.

Let $u = \Omega_u \rightarrow \mathbb{R}$ and $v = \Omega_v \rightarrow \mathbb{R}$ be two given images. We do not assume any global relation between the two images. Let $T_u(x)$ be the structure tensor of u at any point $x \in \Omega_u$ and let $T_v(y)$ be the structure tensor of v at any point $y \in \Omega_v$.

By extending domains of u and v to \mathbb{R}^N (first by an even extension and then by periodicity) and considering structure tensor fields T_u and T_v as metrics on these domains, we obtain two manifolds $(\mathbb{R}^N, G^1 := T_u)$, $(\mathbb{R}^N, G^2 := T_v)$. As was shown in Section 2.2, given two manifolds, an a priori connection

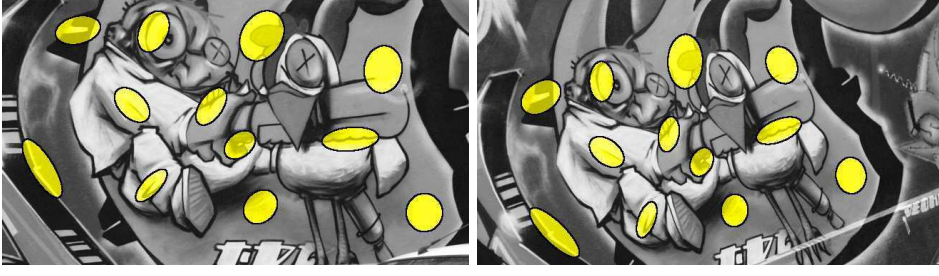


Figure 5.3: Affine covariant neighborhoods (shape-adaptive patches) computed at corresponding points in two images. Despite the difference in viewpoints, patches capture the same visual information.

can be computed as

$$P(x, y) = G^2(y)^{-\frac{1}{2}} G^1(x)^{\frac{1}{2}}.$$

To calculate the square root of $G^1(x)$ and $G^2(y)$ we first diagonalize the tensors matrices

$$T_u(x) = U_u(x) D_u(x) U_u^t(x),$$

$$T_v(y) = U_v(y) D_v(y) U_v^t(y).$$

Here

$$D_u(x) = \text{diag}(\lambda_{u,1}(x), \lambda_{u,2}(x)), \quad \lambda_{u,1}(x) \geq \lambda_{u,2}(x),$$

$$D_v(y) = \text{diag}(\lambda_{v,1}(y), \lambda_{v,2}(y)), \quad \lambda_{v,1}(y) \geq \lambda_{v,2}(y).$$

The matrices $U_u(x)$ and $U_v(y)$ are rotation matrices formed by the eigenvectors of $T_u(x)$ and $T_v(y)$, respectively. Let $e_{u,i}(x)$ be the eigenvector of $T_u(x)$ associated with the eigenvalue $\lambda_{u,i}(x)$, $i \in \{1, 2\}$. Let $e_{v,i}(y)$ be the eigenvector of $T_v(y)$ associated with the eigenvalue $\lambda_{v,i}(y)$. That is, $e_{u,i}(x)$ is the i -th column of $U_u(x)$ and $e_{v,i}(y)$ is the i -th column of $U_v(y)$.

Recall that each structure tensor can be described by its corresponding neighborhood, which in \mathbb{R}^2 is an ellipse given by

$$B_{T_u}(x) = \{\bar{x} : \langle T_u(x)(\bar{x} - x), \bar{x} - x \rangle \leq r^2\},$$

$$B_{T_v}(y) = \{\bar{y} : \langle T_v(y)(\bar{y} - y), \bar{y} - y \rangle \leq r^2\}.$$

Figure 5.3 shows examples of affine covariant neighborhoods computed at corresponding point in two views of the same scene.

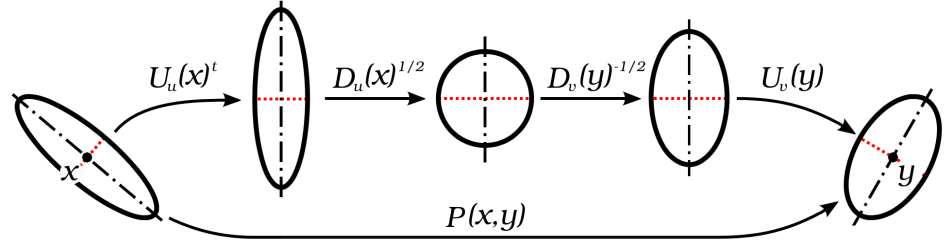


Figure 5.4: Decomposition of an a priori connection $P(x, y)$.

If we define $A_u(x) := D_u(x)^{\frac{1}{2}} U_u(x)^t$, then by the change of variables $X = A_u(x)x'$, we have $A_u(x) \frac{e_{u,i}(x)}{\sqrt{\lambda_{u,i}(x)}} = f_i$, where f_i is a Euclidean orthonormal basis. Which means that $U_u(x)^t$ rotates the ellipse, aligning the minor axis to f_1 and the major to f_2 , and $D_u(x)^{\frac{1}{2}}$ changes the length of both axis. Similarly for the ellipse associated with $T_v(y)$, we define $A_v(y) := D_v(y)^{\frac{1}{2}} U_v(y)^t$ and, by the change of variables $Y = A_v(y)y'$, we have $A_v(y) \frac{e_{v,i}(y)}{\sqrt{\lambda_{v,i}(y)}} = f_i$. After these operations both ellipses are transformed to a standard discs of radius r and hence can be compared. From another point of view, when combined into an a priori connection, these transformations warp an elliptical region at point x into an elliptical region at point y (Figure 5.4).

Using the above notation we can redefine the a priori connection as

$$P(x, y) := A_v(y)^{-1} A_u(x) = U_v(y) D_v(y)^{-\frac{1}{2}} D_u(x)^{\frac{1}{2}} U_u(x)^t. \quad (5.15)$$

Additional Rotation

Similarly to [Garding and Lindeberg \(1994\)](#), it can be shown that in general the formula (5.15) allows to determine local affine transformation from two affine covariant structure tensors, but only up to some orthogonal transformation. The following Lemma 5.4 illustrates this statement. Let us remark that the *exact* local affinity can be obtained by (5.15), when an additional constraint is applied. For example, it is possible in the context of stereo imaging, when the vertical displacement of points is known to be zero after rectification.

For any orthogonal matrix R in \mathbb{R}^2 , let us denote

$$P_R(x, y) = T_v(y)^{-\frac{1}{2}} R T_u(x)^{\frac{1}{2}}.$$

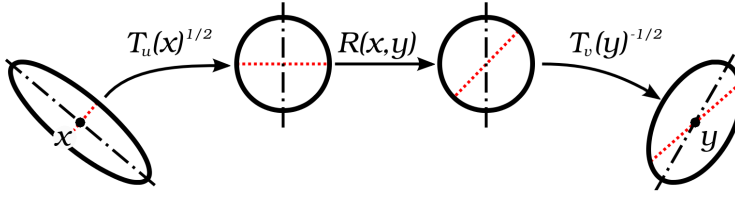


Figure 5.5: Schematic illustration of alignment of elliptical patches that takes into account the additional rotation $R(x, y)$.

Lemma 5.4. *Let u and v be two images, such that $u(z) = v(Az)$ for all $z \in \mathbb{R}^N$ and for some $A \in GL(N)$. Then, $P_R(x, y) = A$ for $y = Ax$ and some orthogonal matrix R .*

Proof. Consider $y = Ax$. To prove $A = P_R(x, y) = T_v(y)^{-\frac{1}{2}} R T_u(x)^{\frac{1}{2}}$ is equivalent to prove that $T_v(y)^{\frac{1}{2}} A T_u(x)^{-\frac{1}{2}}$ is an orthogonal matrix. But T_v is an affine covariant tensor field. Therefore, $T_u(x) = A^t T_v(y) A$. This identity is equivalent to $I = \left(T_v(y)^{\frac{1}{2}} A T_u(x)^{-\frac{1}{2}} \right)^t \left(T_v(y)^{\frac{1}{2}} A T_u(x)^{-\frac{1}{2}} \right)$. That is, $T_v(y)^{\frac{1}{2}} A T_u(x)^{-\frac{1}{2}}$ is an orthogonal matrix. \square

This means that a truly affine invariant a priori connection should be defined as

$$P(x, y) := A_v(y)^{-1} R(x, y) A_u(x) = U_v(y) D_v(y)^{-\frac{1}{2}} R(x, y) D_u(x)^{\frac{1}{2}} U_u(x)^t, \quad (5.16)$$

where $R(x, y)$ is some additional orthogonal transformation. Due to the fact that $U_u(x)$ and $U_v(y)$ are rotations and $D_u(x)^{\frac{1}{2}}$, $D_v(y)^{\frac{1}{2}}$ are scalings, there is no reflection involved in these transformations and, therefore, $R(x, y)$ is indeed a rotation. Figure 5.5 illustrates the complete chain of transformations that align one patch with another.

The additional rotation $R(x, y)$ can be computed from the image content inside elliptical regions at points x and y . When two elliptical regions are normalized to the discs, a proper additional rotation should align these circular regions. Since exhaustive search is not an option for any practical application, we instead split the sought-for rotation into two rotations

$$R(x, y) = R_v(y)^{-1} R_u(x),$$

each of which now depends on image content around only one point, x or y . We look for rotation invariant features inside both normalized circular regions and compute from them two small sets of dominant orientations. Each dominant orientation corresponds to a candidate rotation. Such combination of rotations, that leads to the best alignment, forms the resulting additional rotation $R(x, y)$. In Section 6.3 we explain the estimation of dominant orientations in detail.

Let us finally show that the additional rotation is consistent with our original definition of the a priori connection given in Section 2.2. Recall that from its definition in Section 2.2, the a priori connection is an isometric map that can be generated as (2.9):

$$P(x, y) := B^2(y)Q(x, y)B^1(x)^{-1},$$

where $Q(x, y)$ is any isometry between geodesic coordinate systems around points x and y , $B^1(x)$ and $B^2(y)$ represent some change of coordinate systems, defined by the choice of metrics. For given points x and y , and given metrics, we obtain the whole family of valid a priori connections by varying $Q(x, y)$. For example, if $Q(x, y)$ is the identity, we recover (5.15). The right choice of $Q(x, y)$ allows us to obtain the a priori connection that not only preserves lengths and angles, but also aligns image content in the vicinities of x and y . And the additional rotation $R(x, y)$ in (5.16) is indeed such an isometry.

On the Proposed Affine Invariant Similarity Measures

When $(\mathcal{M}^1, g^1) = (\mathbb{R}^N, T_u)$ and $(\mathcal{M}^2, g^2) = (\mathbb{R}^N, T_v)$, the affine invariance property of the patch similarity measures (4.16) and (4.17) proposed in Chapter 4 stems from the affine covariance property of the proposed structure tensors. In other words, when the proposed affine covariant structure tensors are used as metrics in the image domains, the linear multiscale analysis of similarities is affine invariant.

This fact can also be deduced from the following additional remark on the solutions of equations of the form (4.7). To simplify the notation, let us consider the case of a linear multiscale analysis of images of the same kind. Note that if $A \in GL(N)$ and g is a metric on \mathbb{R}^N defined as above by the affine covariant structure tensor associated with a given image, $u : \mathbb{R}^N \rightarrow \mathbb{R}$, then, as previously noticed, $A : \mathbb{R}^N \rightarrow (\mathbb{R}^N, g)$ induces a metric g_A in \mathbb{R}^N such that $G_A(x) = A^t G(Ax) A$. If $\tilde{\Gamma}$ is the Levi-Civita connection in the

metric g_A , then

$$\tilde{\Gamma}(A^t p) = A^t \Gamma(p) A.$$

Let $(\mathcal{M}^1, g^1) = (\mathbb{R}^N, g)$, $(\mathcal{M}^2, g^2) = (\mathbb{R}^N, g_A)$. Therefore, the equation

$$\frac{\partial u}{\partial t} = \text{Tr}_g (D_{\mathcal{M}^1}^2 u),$$

where $\text{Tr}_g(Q) = \text{Tr}(G^{-1}Q) = g^{ij}Q_{ij}$, is affine invariant. Indeed, since

$$\begin{aligned} \text{Tr}_{g_A} (D_{\mathcal{M}^2}^2 u_A) &= \text{Tr} (G_A^{-1} D_{\mathcal{M}^2}^2 u_A) = \text{Tr} (A^{-1} G^{-1} A^{-t} A^t D_{\mathcal{M}^1}^2 u_A) \\ &= \text{Tr} (G^{-1} D_{\mathcal{M}^1}^2 u A A^{-1}) = \text{Tr} (G^{-1} D_{\mathcal{M}^1}^2 u) = \text{Tr}_g (D_{\mathcal{M}^1}^2 u), \end{aligned}$$

then

$$\frac{\partial}{\partial t} u_A(t, x) = \frac{\partial}{\partial t} u(t, Ax) = \text{Tr}_g (D_{\mathcal{M}^1}^2 u)(t, Ax) = \text{Tr}_{g_A} (D_{\mathcal{M}^2}^2 u_A)(t, x).$$

That is, u_A solves the same equation as $u(t, x)$ with initial condition $u(Ax)$. This gives an example of a linear multiscale analysis that is affine invariant

Notice that the metric depends on the initial condition. This may not be obvious from the notation above, but the equation is applied to a given image, and the metric is constructed from the initial condition. This guarantees that whenever for $u(0, x)$ the metric is g , for $u(0, Ax)$ the metric is $g_A(x) = A^t g(Ax) A$.

5.3 Structure Tensors in Discrete Images

So far we were considering ideal continuous images with infinite resolution. Real images, acquired with a digital camera, are affected by the optical blur (which we assume to be Gaussian) and by the sampling. In this section we analyze, to what extent the affine covariance property of the structure tensors holds for real discrete images. For simplicity, we consider planar images defined on \mathbb{R}^2 . Furthermore as in Morel and Yu (2011), we assume an affine camera model, that is, we disregard perspective effects.

Let $f : \Omega \rightarrow \mathbb{R}$ be an infinite resolution image on a plane, seen from a frontal view. We consider an image $u : \Omega \rightarrow \mathbb{R}$, resulting from looking at f from a different viewpoint. Under an affine camera, we can express u as follows:

$$u(x) = f_A(x) = f(Ax),$$

where $A \in GL(N)$ is an affinity matrix associated with the viewpoint (without loss of generality, we omit the translation of the affinity). Throughout this section we use the notation $f_A = Af$, that is, we use the same symbol for the affinity matrix A and for the operator that warps an image in accordance with affinity A . Since the proposed structure tensors are affine covariant, when computed at corresponding locations on f and u , they should be related by (5.1).

Both u and f are infinite resolution images, seen from two different viewpoints before acquisition. When acquired by a finite resolution camera an image is modified by a filtering operation G_{Σ_1} and a sampling operator S_1 . We assume the G_{Σ_1} to be a Gaussian kernel with covariance matrix Σ_1 , such that its width is the smallest one that allows sampling with a step of 1 without aliasing. We assume that $\Sigma_1 = \sigma_1 I$ as defined in Morel and Yu (2011). Thus, after acquisition we have $\hat{u}_1 = S_1 G_{\Sigma_1} * Af$ and $\hat{f}_1 = S_1 G_{\Sigma_1} * f$. In order to study the effect of blurring separately from the sampling, we consider images $u_1 = G_{\Sigma_1} * Af$ and $f_1 = G_{\Sigma_1} * f$ before sampling.

In general the warping by A and the Gaussian filter do not commute; thus, after acquisition, u_1 and f_1 will not be related by an affinity anymore. Indeed, if G_{Σ} is a Gaussian kernel with covariance matrix Σ and zero mean, then we have that

$$G_{\Sigma} * Af = AG_{A\Sigma A^t} * f,$$

a property which is sometimes referred to as weak commutativity (Morel and Yu (2011)).

Thus, in the affine camera model, a change of viewpoint associated with the affinity A induces an antialiasing Gaussian filter $G_{A\Sigma_1 A^t}$. Structure tensors computed on $u_1 = AG_{A\Sigma_1 A^t} * f_1$ match the ones computed on $G_{A\Sigma_1 A^t} * f_1$, but differ from structure tensors of $f_1 = G_{\Sigma_1} * f$. An exception is given by the case in which the Gaussian kernel is isotropic (as G_{Σ_1}) and $A = R$ is a rotation, since in that case $R\sigma^2 I R^t = \sigma^2 I$. However, this is not true, if the affine transformation involves scalings and/or tilts.

5.4 Empirical Study of the Construction Scheme

The proposed scheme for construction of the affine covariant tensors and neighborhoods is an iterative scheme which starts from some initial region and then alternately updates the structure tensor and the corresponding region. The purpose of the iterative scheme is not to enforce affine covariance property, but rather to diminish dependency on the very first iteration.

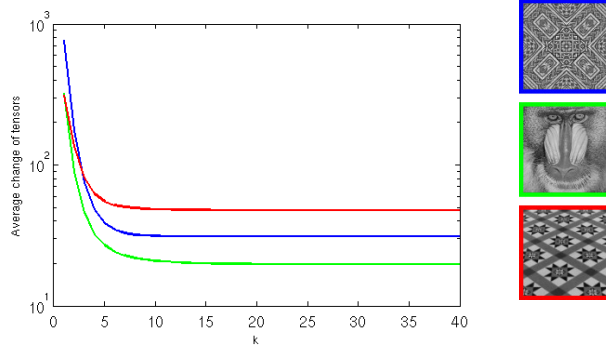


Figure 5.6: Average change of structure tensors from one iteration to another, depending on the number of iteration. Thumbnails of the corresponding images are shown on the right.

By Lemma 5.3 the structure tensor is guaranteed to be affine covariant at any iteration of the scheme. Therefore, we should obtain a correct a priori connection from a pair of structure tensors, if for both of them we run the construction scheme for the same amount of iterations k .

In this section we first study the convergence of the proposed iterative scheme and then also the dependency of structure tensors on the initial iteration for different values of parameter r .

Convergence

Given an image $u : \mathbb{R}^2 \rightarrow \mathbb{R}$, consider the affine covariant tensors $NT^{(k)}(u)(x)$ for $x \in \mathbb{R}^2$, $k \in \mathbb{N}$, $r > 0$. We have empirically observed that after a few iterations the proposed scheme either converges to a single affine covariant structure tensor, or starts to cycle over a finite number of them (typically 2 or 3). Let us note once again, that any of these structure tensors is indeed affine covariant. The convergence can be clarified with the following experiment. Given an image u , we calculate the structure tensors at every point x using different number of iterations k . Then we calculate the Frobenius norms of the differences $\|NT^{(k)}(u)(x) - NT^{(k+1)}(u)(x)\|_F$ for every two consecutive values of k . Finally, we average the norms over all points x in the image u . Figure 5.6 shows these average changes over k for three selected images. The fact that the average changes approach some non-zero value is indeed explained by the occasional alternation between several affine covariant tensors at some points. This behavior is typical for all tested images.

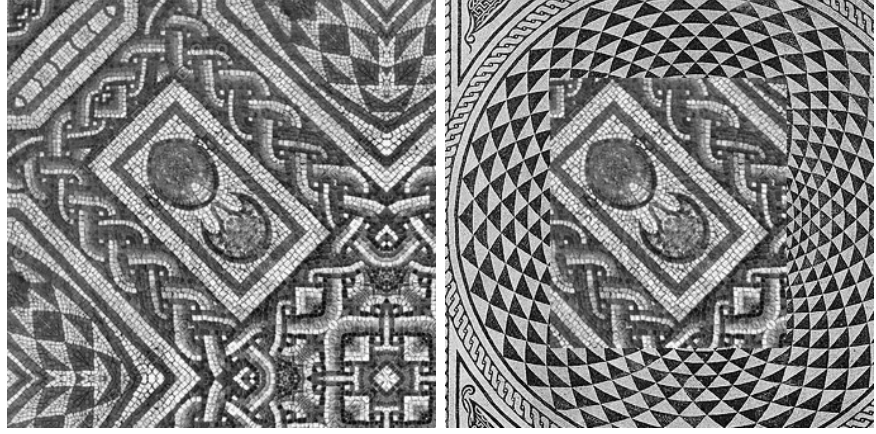


Figure 5.7: Two images with identical central part and different peripheral parts.

In our experiments in this work we always take $k = 30$.

Dependency on Initial Condition

In our iterative construction scheme, the initial region (5.13) is an infinite band whose orientation is given by the gradient at the central point. Therefore, the resulting structure tensor may depend somehow on all the pixels within the initial band. In general this dependency tends to vanish with a sufficient number of iterations of the proposed scheme; however, this tendency is also affected by the image content and the value of r .

In order to study the dependency of the tensors on the initial region we select a highly textured image and create a second image by replacing the peripheral part of the first image with another content (Figure 5.7). Let $T_u(x)$ and $T_v(x)$ be tensors on the first and the second image respectively. We then calculate the error between two corresponding tensors as $e(x) = \|T_u(x) - T_v(x)\|_F^2$, where $\|\cdot\|_F^2$ is the Frobenius norm. Figure 5.8 shows the errors, color-coded by $c(x) = 255 \exp(-\frac{e(x)^2}{2\sigma^2})$, where σ is in the order of 10^2 .

Even for small values of r (50, 100, 150) the majority of points has zero error $e(x)$ and thus does not depend on the initial iteration. However, many covariant neighborhoods (especially close to strong edges) degenerate into rather small sets of pixels or even into a single pixel (Figure 5.9). With such small neighborhoods, the structure tensor estimation may converge to an incorrect solution. Obviously in this case structure tensors do depend

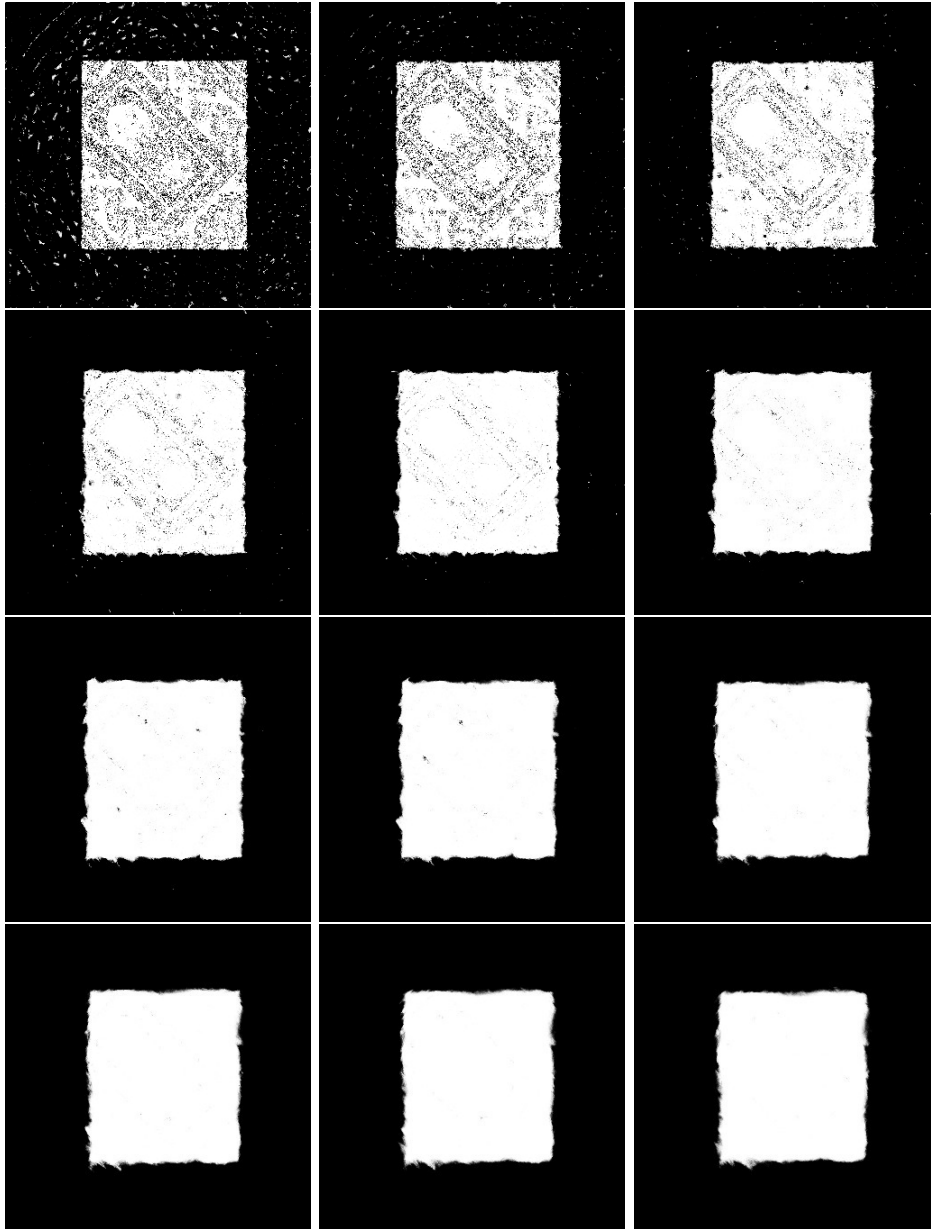


Figure 5.8: Errors between corresponding tensors for different values of r . Darker color means bigger error. From left to right and top to bottom: $r = 50, 100, 150, 200, 250, 300, 350, 400, 450, 500, 550, 600$.

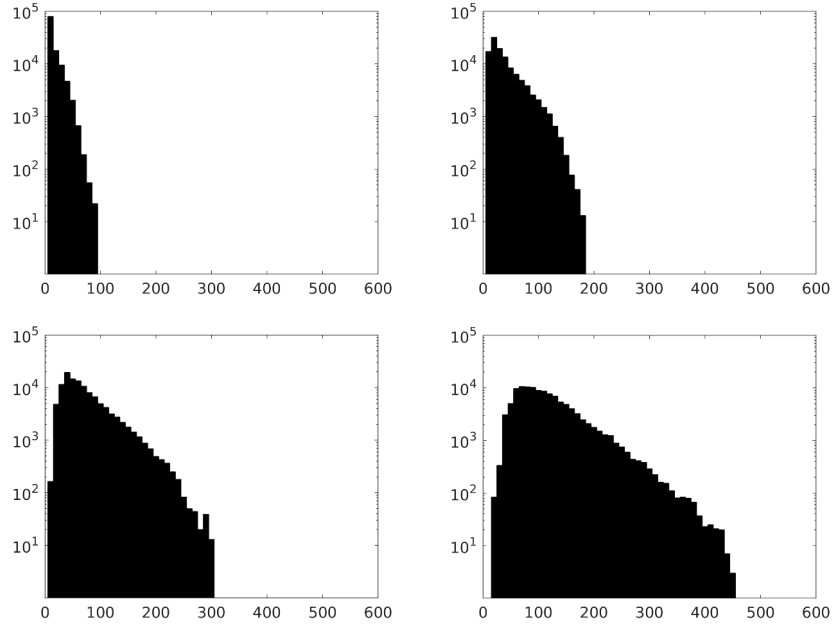


Figure 5.9: Histograms of the ellipse sizes for different values of r . Vertical axis is in logarithmic scale. From left to right and top to bottom: $r = 50, 100, 150, 200$.

on the initial iteration, which explains mismatches in the central part of the images for small r . Moreover, it explains occasional low errors on the peripheral part of the images, where structure tensors close to strong edges occasionally capture the same direction of these edges, but the corresponding neighborhoods do not have enough information to distinguish between two different images.

With the value of r large enough, dependency of structure tensors on the initial iteration is negligible in the whole central region. Let us note that structure tensors near the peripheral region are inevitably influenced by this region in all the iterations and not only during the initial one. Therefore they are irrelevant for this experiment. In Section 7.2, we additionally study the proposed similarity measure on a boundary between two objects, undergoing different transformations.

5.5 Related Approaches to Affine Covariant Regions

The proposed iterative method for the computation of affine covariant structure tensors allows us to estimate the a priori connection between two given points and thus a local affinity. Other methods, suitable for the local affinity estimation, can be found in the literature. Our approach to affine covariant structure tensors and neighborhoods has much in common with the iterative shape adaptation algorithm of Mikolajczyk and Schmid (2004) used in the Harris-Affine feature detector. In this section we compare both approaches in their ability to estimate an affine transformation. We use an implementation of the Harris-Affine feature detector of Vedaldi and Fulkerson (2008).

The method of Mikolajczyk and Schmid (2004) was originally proposed in the feature extraction context for object recognition purposes. It starts by building a scale-space for a given image and detecting stable keypoints together with their characteristic scales. Oriented elliptical neighborhoods are then calculated at these keypoints using an iterative procedure. First, a so-called shape adaptation matrix is estimated in the vicinity of a keypoint, and then the image is warped in accordance with this matrix. The amount of information around a keypoint that is considered in the calculation of the shape adaptation matrix is controlled by the integration scale σ_I . Both σ_I and the position of the keypoint are updated in each iteration. The process is repeated until a convergence criteria is met. In the feature detection context, the integration scale σ_I is related to the characteristic scale of a keypoint.

Like the proposed affine covariant structure tensor, the shape adaptation matrix can be used to estimate an affine transformation that aligns the vicinities of two given points. However, for arbitrary points which are not keypoints, one needs to provide σ_I as a free parameter. In this case parameter σ_I resembles the radius parameter r in our approach. Also, positions of the points should be fixed. Let us note that the method described in Mikolajczyk and Schmid (2004) is initialized on a circular window given by an isotropic Gaussian which is not affine covariant. Whereas, our scheme begins from an affine covariant initial band and guarantees that at any stage of the algorithm the computed tensor is affine covariant.

In order to compare both methods we use an image sequence well known in the feature detection community: the *graffiti* sequence, taken from the test data in Mikolajczyk (2007). It contains six images showing different views

of the same scene. Five ground truth global transformations from the first image to all the others are known. We uniformly sample 2745 points in the first image and use the ground truth transformations to obtain the sets of corresponding points for the other images. We then estimate local affinities $A(x, y)$ for every pair of corresponding points x and y using both methods. Knowing locations of the corresponding points we obtain estimated global transformations $\bar{A}(x, y)$ from local ones. For that we write the transformations in the 2D projective space using homogeneous coordinates, and obtain

$$\bar{A}(x, y) := \begin{bmatrix} I & y \\ 0 & 1 \end{bmatrix} \begin{bmatrix} A(x, y) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} I & -x \\ 0 & 1 \end{bmatrix}$$

where $x, y \in \mathbb{R}^2$. We measure the estimation error by the Frobenius norm of the difference between the estimated and the ground truth transformations. Let us note once again that in both methods the sizes of neighborhoods are controlled by free parameters: r in our method and σ_I in the Harris-Affine one. However, the values of r and σ_I could not be easily related. Therefore, for each pair of images we test extensive ranges of r and σ_I and select the ones giving the smallest median value of the error. With this experiment we indirectly evaluate the degree of affine covariance of the proposed covariant structure tensors and affine normalization proposed in Mikolajczyk and Schmid (2004). Figure 5.10 shows statistics for both methods over the five pairs of images. It can be seen that statistically our method performs slightly better.

Let us note that other approaches to affine neighborhoods exist, for example, Baumberg (2000); Tuytelaars and Van Gool (2004); Matas et al. (2004); Morel and Yu (2009). However, using these methods it is impractical or even impossible to extract affine covariant regions densely. In contrast, our approach by design is capable of producing dense field of affine covariant neighborhoods.

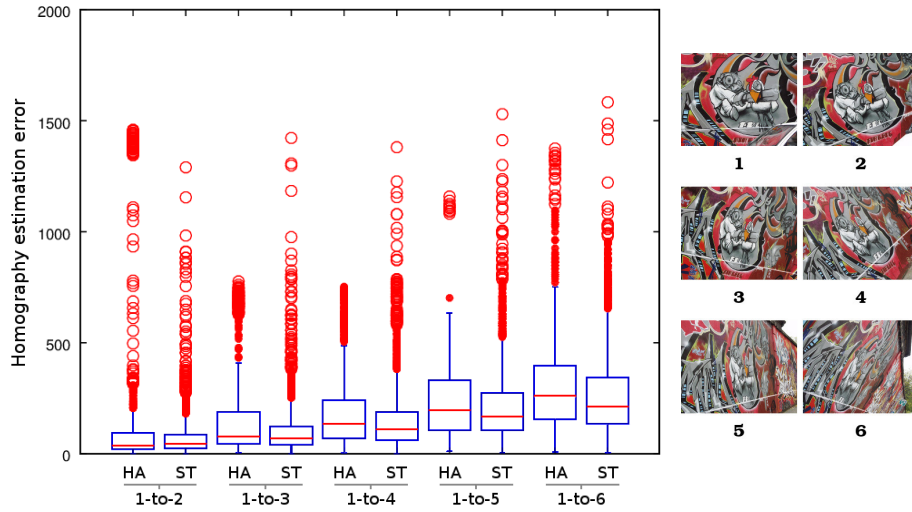


Figure 5.10: Boxplot showing statistics of homography estimation error for 2745 uniformly sampled points. Homography is estimated for the first image and all the others (labeled as “1-to-2”, “1-to-3” and so on). Estimation using shape adaptation matrix of Harris-Affine feature detector (labeled as “HA”) is compared with our approach using structure tensors (labeled as “ST”). As usual, boxes represent the first, second and third quartiles, the length of the whiskers corresponds to 1.5 interquartile range (IQR) and red dots are outliers. Thumbnails of the images being used are shown on the right.

Numerical Implementation

In this chapter we describe a numerical implementation of the proposed affine invariant patch similarity measures (4.16) and (4.17) that we use in all experiments and applications throughout this work. Recall that these similarity measures involve Riemannian metrics, and that the affine covariant structure tensors, studied in Chapter 5, can be used as such metrics. Both similarity measures can be written as

$$\mathcal{D}^a(t, x, y) = \int_{\Delta_t} g_t(h) \left\| u(x + T_u^{-\frac{1}{2}}(x)R_u^{-1}(x)h) - v(y + T_v^{-\frac{1}{2}}(y)R_v^{-1}(y)h) \right\|_2^2 dh, \quad (6.1)$$

where $g_t(h)$ is either a Gaussian of variance t , or an approximated geodesic weighting function, T_u and T_v are the structure tensor fields, R_u and R_v are the additional rotations. In this dissertation we always refer to u and v as *images*, even though they might as well represent regular videos, sequences of 3D images captured over time, etc. In general, we say that $u : \Omega_u \subset \mathbb{R}^N \rightarrow \mathbb{R}^M$ and $v : \Omega_v \subset \mathbb{R}^N \rightarrow \mathbb{R}^M$ are defined on \mathbb{R}^N with values in \mathbb{R}^M . However, throughout this chapter we consider only two most practical cases, namely $N = 2, 3$. In \mathbb{R}^3 , u and v might represent regular videos or 3D images. The number of color channels M is usually assumed to be 1 or 3. Since the particular choice of M does not affect the reasoning in this chapter, we do not specify it explicitly. Without loss of generality we assume that $u = v$.

The similarity measure (6.1) assigns a distance to a pair of shape-adaptive patches centered at two given points in the following way. The scheme described in Section 5.1 allows to calculate at any given point x both the affine

covariant structure tensor and the affine covariant region (shape-adaptive patch) defined by it. In order to compare two shape-adaptive patches we have to align (register) them first. A proper registration can be obtained from the structure tensors; however, as commented in Section 5.2, we have to specify additional rotations. We estimate these missing rotations by extracting dominant orientations of gradients within the patches. For the purposes of comparison we normalize shape-adaptive patches to discs or spheres of the same radius and interpolate these normalized versions to the regular grid. In this way we obtain a convenient representation of shape- and size-varying patches by arrays of interpolated color values whose sizes are all equal and known in advance. Notice that there might be several dominant orientations within a patch; therefore, for a single point x we might have multiple normalized versions of the patch. We compare every version associated with the point x with every version associated with the point y and finally assign to these points the smallest distance among all the combinations.

In the following section we present a more formal overview of the numerical implementation and then describe specific parts of it in more detail.

6.1 Outline of Patch Similarity Computation

This section presents an overview of the numerical implementation of the proposed similarity measures. The following high-level outline of the algorithm is structured in the form of a data flow, where every step is described by a set of inputs (“in”), a set of internal parameters (“prm”) and a set of outputs (“out”). Figure 6.1 shows a data flow diagram that graphically represents the outline. Locations x and y are the inputs of the algorithm itself. They can come from the same image u or from two different images u and v ; however, for simplicity of notation we assume here that $u = v$. Distance d , together with the corresponding configuration of additional rotations, are the outputs of the algorithm.

Let us recall some useful notation from the previous chapters. We denote by u a given image and by ∇u its gradient. We denote by $T_u(x)$ the affine covariant structure tensor at point x and by $B_{T_u}(x)$ its corresponding affine covariant neighborhood. Parameter r controls the size of affine covariant neighborhoods in the construction scheme described in Section 5.1. The rest of the notation used in the outline is explained upon appearance.

Outline

0. Build regular grid for interpolation

prm: g, r
 out: $\mathcal{G} := \{w_i\}$

Parameter g controls the resolution of the regular grid, set \mathcal{G} contains coordinates of the grid nodes. This is a preprocessing step that normally should be done only once and before any patch distance computations. See Section 6.4 for details.

1. Calculate structure tensor and shape-adaptive patch

in: $\nabla u, x$
 prm: r, n_{ST}
 out: $T_u(x), B_{T_u}(x)$

Parameter n_{ST} controls the number of iterations in the construction scheme for the affine covariant structure tensors and neighborhoods (Section 5.1). The algorithm for collecting points belonging to the neighborhood $B_{T_u}(x)$ is explained in Section 6.2.

2. Estimate dominant orientations

in: $\nabla u, B_{T_u}(x), T_u(x), x$
 prm: $n_{bins}, \sigma_{DO}, \delta, n_{DO}$
 out: $\{\Theta_k\}$

Parameter n_{bins} controls the number of bins in the gradient orientations histogram, parameter σ_{DO} controls the intra-patch weighting and parameter δ is the cut-off threshold for local maxima as in Lowe (2004). Parameter n_{DO} limits the maximum number of orientations to output and $\{\Theta_k\}$ is the set of estimated dominant orientations. See Section 6.3 for details.

3. Normalize shape-adaptive patch (for each Θ_k)

in: $u, T_u(x), B_{T_u}(x), x, \Theta_k$
 out: $\bar{B}_k(x) := \{(z_j, \bar{z}_j, c_j)\}$

Set $\bar{B}_k(x)$ contains points of the neighborhood $B_{T_u}(x)$, transformed by $R(\Theta_k)T_u(x)^{\frac{1}{2}}$, where $R(\Theta_k)$ represents an additional rotation. Every j -th element of the set $\bar{B}_k(x)$ includes original coordinate z_j , transformed coordinate \bar{z}_j and color value $c_j = u(z_j)$. See Section 6.4 for details.

4. Interpolate normalized patch to the grid \mathcal{G} (for each Θ_k)

in: $\bar{B}_k(x), \mathcal{G}, \sigma_{\mathcal{G}}$

out: $\{\bar{c}_i\}_k$

Here $\sigma_{\mathcal{G}}$ is the interpolation parameter. Output set contains interpolated color values. For every node w_i of the regular grid \mathcal{G} we obtain the interpolated color value \bar{c}_i . See Section 6.4 for details.

Note: after this step all interpolated candidate normalizations, corresponding to the same point x , are combined in a set $\mathcal{P}(x) := \{(\{\bar{c}_i\}_k, \Theta_k)\}$.

5. Calculate patch distance between points x and y

in: $\mathcal{P}(x), \mathcal{P}(y)$

out: d, Θ_x, Θ_y

Every candidate normalization from $\mathcal{P}(x)$ is compared with every candidate normalization from $\mathcal{P}(y)$, and a configuration that gives the smallest distance is returned.

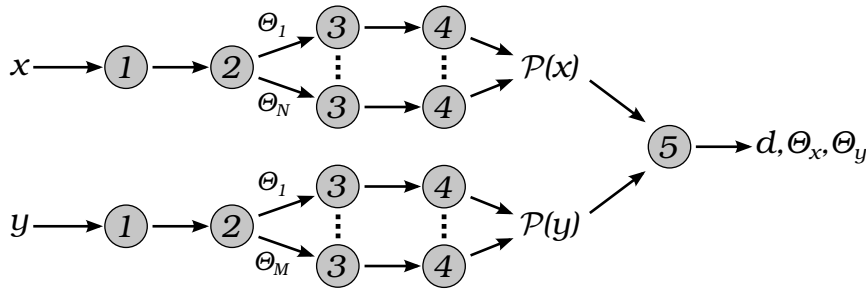


Figure 6.1: Data flow diagram for the patch similarity (distance) calculation between given points x and y . Numbers in the nodes correspond to the steps of the outline.

6.2 Affine Covariant Regions

The numerical scheme for the construction of affine covariant structure tensors and neighborhoods is described in Section 5.1. In this section we explain an efficient way to determine a set of points that belong to a given affine covariant region. A naïve approach would be to apply the *flood fill* algorithm that is used, for example, in image editing software to fill connected areas of

the same color with a new color. Given a seed point the *flood fill* recursively moves from point to point and fills a region from the inside out. Luckily in our case the shape of a region is completely defined by its corresponding structure tensor. This leads to a much simpler algorithm without recursion. At first we consider the \mathbb{R}^2 case, when affine covariant regions are ellipses, and then we extend the algorithm to \mathbb{R}^3 for processing ellipsoids.

Throughout this section we use different notation for points in images to simplify explicit coordinates indication. Let $\bar{\mathbf{p}}$ be the central point of an affine covariant region $B_{T_u}(\bar{\mathbf{p}})$ and \mathbf{p} be just any point on u . Point \mathbf{p} belongs to an affine covariant region $B_{T_u}(\bar{\mathbf{p}})$ if and only if

$$\langle T_u(\bar{\mathbf{p}})(\mathbf{p} - \bar{\mathbf{p}}), (\mathbf{p} - \bar{\mathbf{p}}) \rangle \leq r^2$$

Then the boundary of an affine covariant region $B_{T_u}(\bar{\mathbf{p}})$ is given by

$$\partial B_{T_u}(\bar{\mathbf{p}}) = \langle T_u(\bar{\mathbf{p}})(\mathbf{p} - \bar{\mathbf{p}}), (\mathbf{p} - \bar{\mathbf{p}}) \rangle = r^2. \quad (6.2)$$

Affine Covariant Regions in \mathbb{R}^2

Let us indicate components of $\bar{\mathbf{p}} \in \mathbb{R}^2$ and $\mathbf{p} \in \mathbb{R}^2$ as $\bar{\mathbf{p}} = [\bar{x}, \bar{y}]^t$ and $\mathbf{p} = [x, y]^t$. From the expression (6.2) we can compute two points on $\partial B_{T_u}(\bar{\mathbf{p}})$ with the biggest and the smallest y coordinates. Let

$$T_u(\bar{\mathbf{p}}) = \begin{bmatrix} T_{00} & T_{01} \\ T_{10} & T_{11} \end{bmatrix},$$

where $T_{01} = T_{10}$. By taking partial derivative of (6.2) with respect to x we obtain

$$T_{00}(x - \bar{x}) + T_{01}(y - \bar{y}) + T_{01}(x - \bar{x})\frac{\partial y}{\partial x} + T_{11}(y - \bar{y})\frac{\partial y}{\partial x} = 0.$$

Then by setting $\frac{\partial y}{\partial x} = 0$ we have

$$x - \bar{x} = -\frac{T_{01}}{T_{00}}(y - \bar{y}).$$

By substituting the above expression into (6.2) we obtain the y coordinates of the two extreme points (shown in Figure 6.2a)

$$y = \bar{y} \mp r \left(T_{11} - \frac{T_{01}^2}{T_{00}} \right)^{-\frac{1}{2}}. \quad (6.3)$$

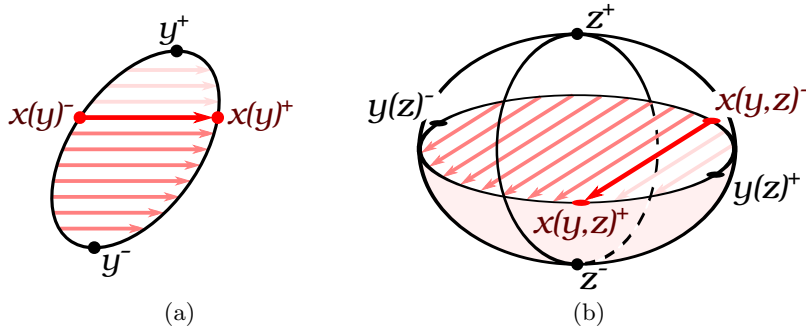


Figure 6.2: Schematic illustration of traversing of shape-adaptive patches. (a) elliptical patch in \mathbb{R}^2 . (b) ellipsoidal patch in \mathbb{R}^3 .

Let us denote them y^- and y^+ . To collect the points \mathbf{p} that belong to $B_{T_u}(\bar{\mathbf{p}})$ we traverse rows lying between the extreme points. For every row y (where $y^- \leq y \leq y^+$) we compute the x coordinates of two its intersections with the elliptical boundary (6.2) as

$$x(y) = \bar{x} - a(y - \bar{y}) \pm \sqrt{b(y - \bar{y})^2 + c}, \quad (6.4)$$

where the constants are given by

$$a = \frac{T_{01}}{T_{00}}, \quad b = a^2 - \frac{T_{11}}{T_{00}} \quad \text{and} \quad c = \frac{r^2}{T_{00}}.$$

Then we collect all points \mathbf{p} belonging to that row and located within the boundary of $B_{T_u}(\bar{\mathbf{p}})$. Figure 6.2a schematically illustrates the traversing.

At some points near high contrast edges, for small values of r , the numerical scheme may yield degenerate structure tensors which are either not positive definite, or close to it and result in extremely elongated and narrow regions. In both cases we output a region containing only the central point $\bar{\mathbf{p}}$ instead. To check that a matrix is positive definite we check that $\det(A) > 0$ and $A_{00} > 0$. For the structure tensor, T_{00} is always greater then or equal to 0; thus it is sufficient to check only its determinant. The elongation of an elliptical region can be found using the ratio between eigenvalues of the corresponding structure tensor. As discussed in Lowe (2004) (Section 4.1), since only the ratio is needed, there is no need to compute the eigenvalues themselves. Let λ_1 and λ_2 be eigenvalues of a structure tensor $T_u(\mathbf{p})$. Then

its trace and determinant are

$$\begin{aligned}\text{Tr}(T_u(\mathbf{p})) &= \lambda_1 + \lambda_2, \\ \text{Det}(T_u(\mathbf{p})) &= \lambda_1 \lambda_2.\end{aligned}$$

Let $\alpha = \lambda_1/\lambda_2$ be the eigenvalue ratio. Then

$$\frac{\text{Tr}(T_u(\mathbf{p}))^2}{\text{Det}(T_u(\mathbf{p}))} = \frac{(\lambda_1 + \lambda_2)^2}{\lambda_1 \lambda_2} = \frac{(\alpha \lambda_2 + \lambda_2)^2}{\alpha \lambda_2^2} = \frac{(\alpha + 1)^2}{\alpha}.$$

Notice that the previous expression, $f(\alpha) := \frac{(\alpha+1)^2}{\alpha}$, is an increasing function of α . Therefore, when

$$\frac{\text{Tr}(T_u(\mathbf{p}))^2}{\text{Det}(T_u(\mathbf{p}))} > \frac{(\bar{\alpha} + 1)^2}{\bar{\alpha}}$$

for a given eigenvalue ratio threshold $\bar{\alpha}$, we recognize the structure tensor $T_u(\mathbf{p})$ as degenerate as well. Throughout this work we set $\bar{\alpha} = 100$.

The whole procedure is summarized in Algorithm 1.

Affine Covariant Regions in \mathbb{R}^3

Let us indicate components of $\bar{\mathbf{p}} \in \mathbb{R}^3$ and $\mathbf{p} \in \mathbb{R}^3$ as $\bar{\mathbf{p}} = [\bar{x}, \bar{y}, \bar{z}]^t$ and $\mathbf{p} = [x, y, z]^t$. When the video case is considered, we can think of z as the time coordinate.

In \mathbb{R}^3 every affine covariant region $B_{T_u}(\bar{\mathbf{p}})$ is an ellipsoid. To efficiently collect points, that belong to $B_{T_u}(\bar{\mathbf{p}})$, we cut it into \mathbb{R}^2 slices in some direction and then assembly it slice by slice. The slices have elliptical shape defined by the corresponding structure tensor $T_u(\bar{\mathbf{p}})$; therefore we can apply almost the same procedure as before to process them. Let

$$T_u(\bar{\mathbf{p}}) = \begin{bmatrix} T_{00} & T_{01} & T_{02} \\ T_{01} & T_{11} & T_{12} \\ T_{02} & T_{12} & T_{22} \end{bmatrix}.$$

In the video case it would be natural to cut $B_{T_u}(\bar{\mathbf{p}})$ by video frames; thus, we choose to do so in the direction of z . Notice, however, that the direction of cutting only affects indexing. We can determine the first and the last slices of u , containing points of $B_{T_u}(\bar{\mathbf{p}})$, by taking partial derivatives of (6.2) with respect to x and y , and setting them to zero:

$$\begin{cases} T_{00}(x - \bar{x}) + T_{01}(y - \bar{y}) + T_{02}(z - \bar{z}) = 0 \\ T_{01}(x - \bar{x}) + T_{11}(y - \bar{y}) + T_{12}(z - \bar{z}) = 0 \end{cases}.$$

Algorithm 1: Assembly of affine covariant regions in \mathbb{R}^2 .

Input: $T_u(\bar{\mathbf{p}})$, Ω_u , $\bar{\mathbf{p}} = [\bar{x}, \bar{y}]^t$ // $\Omega_u \in \mathbb{R}^2$ is domain of u .
Parameters: r , $\bar{\alpha}$
Output: $B_{T_u}(\bar{\mathbf{p}})$
 $T := T_u(\bar{\mathbf{p}})$ // alias
 $B_{T_u}(\bar{\mathbf{p}}) \leftarrow \{\}$
 // Ensure tensor in not degenerate
if $\det(T) \leq 0$ **or** $\frac{\text{trace}(T)^2}{\det(T)} > \frac{(\bar{\alpha}+1)^2}{\bar{\alpha}}$ **then**
 $B_{T_u}(\bar{\mathbf{p}}) \leftarrow B_{T_u}(\bar{\mathbf{p}}) \cup \bar{\mathbf{p}}$
 Stop
 // Compute offsets of extreme points from the center
 $oy \leftarrow r \left(T_{11} - \frac{T_{01}^2}{T_{00}} \right)^{-\frac{1}{2}}$ // Eq. (6.3)
 // Compute auxiliary constants
 $a \leftarrow \frac{T_{01}}{T_{00}}$
 $b \leftarrow a^2 - \frac{T_{11}}{T_{00}}$
 $c \leftarrow \frac{r^2}{T_{00}}$
 // Traverse row by row
for $\bar{y} - \lfloor oy \rfloor \leq y \leq \bar{y} + \lfloor oy \rfloor$ **do** // $y \in \mathbb{R}$
 Skip y , if it is outside of Ω_u
 $x^- \leftarrow \bar{x} - a(y - \bar{y}) - \sqrt{b(y - \bar{y})^2 + c}$ // left intersection
 $x^+ \leftarrow \bar{x} - a(y - \bar{y}) + \sqrt{b(y - \bar{y})^2 + c}$ // right intersection
 for $\lceil x^- \rceil \leq x \leq \lfloor x^+ \rfloor$ **do**
 $B_{T_u}(\bar{\mathbf{p}}) \leftarrow B_{T_u}(\bar{\mathbf{p}}) \cup \mathbf{p}$ // $\mathbf{p} = [x, y]^t$

From the above system and (6.2) we obtain that

$$z = \bar{z} \pm r \left(T_{00}b^2 + T_{11}a^2 + T_{22} + 2T_{01}ab + 2T_{02}b + 2T_{12}a \right)^{-\frac{1}{2}},$$

where the constants are given by

$$a = \frac{T_{01}T_{02} - T_{00}T_{12}}{T_{00}T_{11} - T_{01}^2} \quad \text{and} \quad b = -\frac{1}{T_{00}}(T_{01}a + T_{02}).$$

Let us denote these coordinates as z^- and z^+ . For every slice z (where $z^- \leq z \leq z^+$) within the ellipsoid we apply a procedure similar to the one

explained in Algorithm 1 to collect all points belonging to that slice. The main difference is that instead of the equation (6.3) for the y coordinates of extreme points we now have

$$y(z) = \bar{y} + a(z - \bar{z}) \pm \sqrt{b(z - \bar{z})^2 - c}, \quad (6.5)$$

where the constants are given by

$$a = \frac{T_{01}T_{02} - T_{00}T_{12}}{T_{00}T_{11} - T_{01}^2}, \quad b = a^2 - \frac{T_{00}T_{22} - T_{02}^2}{T_{00}T_{11} - T_{01}^2} \quad \text{and} \quad c = \frac{T_{00}r^2}{T_{00}T_{11} - T_{01}^2}.$$

Instead of the equation (6.4) for the intersections of rows of slices with the boundary of $B_{T_u}(\bar{\mathbf{p}})$ we have slightly more complicated

$$x(y, z) = \bar{x} - a(y - \bar{y}) - b(z - \bar{z}) \pm \sqrt{c(y - \bar{y})^2 + d(z - \bar{z})^2 + e(y - \bar{y})(z - \bar{z}) + f}, \quad (6.6)$$

where the constants are given by

$$a = \frac{T_{01}}{T_{00}}, \quad b = \frac{T_{02}}{T_{00}}, \quad c = a^2 - \frac{T_{11}}{T_{00}}, \quad d = b^2 - \frac{T_{22}}{T_{00}},$$

$$e = ab - \frac{T_{12}}{T_{00}}, \quad f = \frac{r^2}{T_{00}}.$$

Figure 6.2b schematically illustrates the traversing in the \mathbb{R}^3 case.

6.3 Dominant Orientations

In Section 5.2 it was shown that a local affine transformation can be estimated from two structure tensors, but only up to a rotation. To compensate for the missing rotation we estimate dominant orientations of the normalized patches using histograms of gradient orientations as in the SIFT descriptors of Lowe (2004). Notice that there might be several dominant orientations and thus several equivalent options for the additional rotation. The procedure for estimating dominant orientations is described below and summarized in Algorithm 2 for the case of images in \mathbb{R}^2 . Thereafter an extension for the \mathbb{R}^3 case is also proposed.

Dominant Orientations in \mathbb{R}^2

Recall that in order to compute affine covariant structure tensors we first compute the gradient field ∇u . The same gradient vectors, when transformed appropriately, can be used to estimate the dominant orientation within normalized patches. Let $\hat{T}(x) = T_u(x)^{\frac{1}{2}}$ be the transformation that normalizes an elliptical patch at x to a disc of radius r . Of course, this normalization does not yet take any additional rotation into account. It can be shown that the suitable transformation to apply to the gradient vectors is $(\hat{T}(x)^{-1})^t$.

For every point $y \in B_{T_u}(x)$ we transform the corresponding gradient vector $\nabla u(y)$, compute its direction and magnitude and use them to fill-in the circular histogram of orientations. Note that the magnitude value is additionally weighted by the anisotropic Gaussian intra-patch weight, depending on the distance to the center of the patch, given by

$$\omega_x(y) = \exp\left(-\frac{\langle T_u(x)(y-x), (y-x) \rangle}{2\sigma_{DO}^2}\right). \quad (6.7)$$

The resulting value is then distributed linearly between the nearest bins of the histogram in proportion to the distance to these bins. In that histogram we find the global maximum and all other local maxima which are big enough. We fit the quadratic function to the histogram values around every such maximum and take its argmax as orientation Θ in order to improve the accuracy of estimation. Up to n_{DO} highest peaks in the histogram are considered as dominant orientations and gathered into the output set $\{\Theta_k\}$. According to our observations, most commonly the number of dominant orientations does not exceed two.

Given a dominant orientation Θ , the additional rotation $R_u(x)$ can be computed as

$$R_u(x) := R(\Theta) = \begin{bmatrix} \cos \Theta & \sin \Theta \\ -\sin \Theta & \cos \Theta \end{bmatrix}. \quad (6.8)$$

In our experiments we set the number of bins in the histogram $n_{bins} = 72$, the cut-off threshold $\delta = 45\%$, the weighting parameter $\sigma_{DO} = 0.5$ and the limit on the number of orientations $n_{DO} = 3$.

Dominant Orientations in \mathbb{R}^3

In contrast to the \mathbb{R}^2 case, where a single angle fully defines an orientation, in \mathbb{R}^3 we need three angles: two angles to define a *direction* and a tilt

Algorithm 2: Estimation of dominant orientations in \mathbb{R}^2 .

Input: $\nabla u, B_{T_u}(x), T_u(x), x \in \mathbb{R}^2$ **Parameters:** $n_{bins}, \sigma_{DO}, \delta, n_{DO}$ **Output:** $\{\Theta_k\}$

```

 $\sigma := \sigma_{DO}$  // alias
 $A \leftarrow ((T_u(x)^{\frac{1}{2}})^{-1})^t$  // transformation for gradient vectors
 $H \leftarrow$  array of  $(n_{bins} + 2)$  elements, all set to 0 // histogram

// Fill-in histogram  $H$ 
foreach  $y \in B_{T_u}(x)$  do
     $\widetilde{\nabla}u(y) \leftarrow A\nabla u(y)$ 
     $\alpha \leftarrow \exp\left(-\frac{\langle T_u(x)(y-x), (y-x) \rangle}{2\sigma^2}\right) \|\widetilde{\nabla}u(y)\|$ 
     $\gamma \leftarrow$  angle between  $\widetilde{\nabla}u(y)$  and  $X$  axis in range  $[0, 2\pi]$ 
     $p \leftarrow \frac{\gamma n_{bins}}{2\pi}$  // real-valued position in histogram
     $i \leftarrow \lfloor p - 0.5 \rfloor + 1$  // index of the left closest bin
     $d \leftarrow p - i + 0.5$  // distance to the center of  $i$ -th bin
     $H_i \leftarrow H_i + (1 - d) \alpha$ 
     $H_{i+1} \leftarrow H_{i+1} + d \alpha$ 

// Merge boundary values, because  $H$  should be circular
 $H_0 \leftarrow H_0 + H_{-2}$  //  $H_{-2}$  is penultimate element
 $H_{-1} \leftarrow H_{-1} + H_1$  //  $H_{-1}$  is last element
 $H_{-2} \leftarrow H_0$ 
 $H_1 \leftarrow H_{-1}$ 

Smooth  $H$  by convolving it six times with  $[\frac{1}{3}, \frac{1}{3}, \frac{1}{3}]$  kernel

// Collect peaks that are big enough
 $C \leftarrow \{\}$  // set of candidate orientations
for  $1 \leq i \leq n_{bins}$  do
    if  $H_i \geq \delta \max(H)$  and  $H_i > H_{i-1}$  and  $H_i > H_{i+1}$  then
         $C \leftarrow C \cup (H_i, i)$  // append tuple  $(H_i, i)$  to  $C$ 

Sort  $C$  in descending order by histogram values

// Refine and collect dominant orientations
 $\Theta \leftarrow \{\}$ 
for  $0 \leq j < \min(n_{DO}, \text{size}(C))$  do
     $i \leftarrow$  second element from tuple  $C_j$ 
     $\Theta \leftarrow \Theta \cup \frac{2\pi}{n_{bins}} \left( i + 0.5 + 0.5 \frac{H_{i-1} - H_{i+1}}{H_{i-1} - 2H_i + H_{i+1}} \right)$ 

```

angle around that direction to completely define an orientation. By analogy with the geographic coordinate system, we can call the first two angles longitude and latitude. Dominant orientations can still be computed using a slightly modified approach of SIFT based on the histograms of gradient orientations. As before, given an ellipsoidal affine covariant region $B_{T_u}(x)$, we first transform the gradient vectors within it by $(T_u(x)^{-\frac{1}{2}})^t$ and obtain a set of vectors within a sphere.

For every transformed gradient vector we compute its direction (first two angles) and magnitude. Figure 6.3a illustrates the two angles defining direction. In that setting angle $\theta \in [0, 2\pi]$ denotes *longitude* and angle $\varphi \in [0, \pi]$ denotes *colatitude*, which is the complementary angle of latitude. We fill-in the 2D histogram of directions with the magnitudes of gradients according to their θ and φ . Notice that the histogram should be circular in the longitude dimension. As in the \mathbb{R}^2 case, the magnitudes are weighted depending on the distance to the center of the patch using (6.7). Each contribution should additionally be normalized by the area (or solid angle) of its corresponding bin, because bins near the equator of a sphere are larger than at its poles and this introduces an unwanted bias. The solid angle can be computed as

$$\omega(\theta, \varphi) = \Delta\theta(\cos(\varphi) - \cos(\varphi + \Delta\varphi)),$$

where $\Delta\theta = 2\pi/n_{bins}$ and $\Delta\varphi = 2\pi/n_{bins}$ stands for the span of the bins. The complete weight for every contribution at $y \in B_{T_u}(x)$ is then given by

$$\omega(y) := \frac{\omega_x(y)}{\omega(\theta_y, \varphi_y)}.$$

After filling-in and smoothing the 2D histogram we locate at most n_{DO} peaks of dominant *directions*. To refine values of θ and φ we fit the quadratic function to the histogram values around the peaks separately for each of two angles.

For every dominant direction (θ_i, φ_i) we now need to compute a tilt angle. Let us call a plane orthogonal to the direction (θ_i, φ_i) a *tilt plane*. The idea is to apply the similar procedure as in the \mathbb{R}^2 case to the orthogonal projections of the transformed gradient vectors onto a tilt plane. For that we first compute a rotation that aligns (θ_i, φ_i) direction with the Z axis

$$R(\theta, \varphi) = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta \cos\varphi & \cos\theta \cos\varphi & -\sin\varphi \\ \sin\theta \sin\varphi & \cos\theta \sin\varphi & \cos\varphi \end{bmatrix}.$$

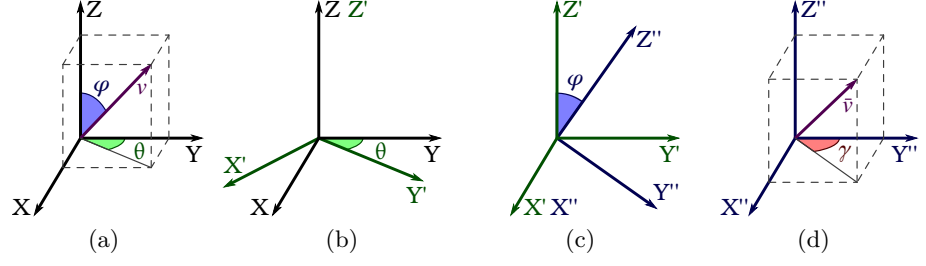


Figure 6.3: Orientations in \mathbb{R}^3 . (a) longitude θ and colatitude φ of some vector v . (b) rotation of the coordinate system around Z axis by θ . (c) rotation of the coordinate system around X' axis by φ . (d) tilt γ of some other vector \bar{v} in the rotated coordinate system.

This rotation can be decomposed into two steps. At the first step the original coordinate system XYZ is rotated around Z axis by θ which gives the coordinate system $X'Y'Z'$ (Figure 6.3b). At the second step the coordinate system $X'Y'Z'$ is rotated around X' axis by φ which gives the coordinate system $X''Y''Z''$ (Figure 6.3c). In this new coordinate system $X''Y''Z''$ the plane $X''Y''$ corresponds to a tilt plane. Figure 6.3d illustrates the tilt angle $\gamma \in [0, 2\pi]$. To compute dominant tilts we rotate every transformed gradient vector by $R(\theta_i, \varphi_i)$, compute its tilt angle γ , compute the magnitude of its projection onto the coordinate plane $X''Y''$ and proceed with the histogram of tilt angles as in the \mathbb{R}^2 case. As previously, up to n_{DO} tilt angles can be obtained for every pair (θ_i, φ_i) ; thus, the explained procedure might return at most $(n_{DO})^2$ dominant orientations $\Theta = (\theta, \varphi, \gamma)$ in the \mathbb{R}^3 case.

Given all three angles θ , φ and γ that in \mathbb{R}^3 constitute a single dominant orientation Θ , the additional rotation $R_u(x)$ can finally be computed as

$$R_u(x) := R(\Theta) = R(\theta, \varphi, \gamma) = \begin{bmatrix} \cos\theta \cos\gamma - \sin\theta \cos\varphi \sin\gamma & -\sin\theta \cos\gamma - \cos\theta \cos\varphi \sin\gamma & \sin\varphi \sin\gamma \\ \cos\theta \sin\gamma + \sin\theta \cos\varphi \cos\gamma & -\sin\theta \sin\gamma + \cos\theta \cos\varphi \cos\gamma & -\sin\varphi \cos\gamma \\ \sin\theta \sin\varphi & \cos\theta \sin\varphi & \cos\varphi \end{bmatrix}. \quad (6.9)$$

6.4 Normalization and Interpolation

In this section we delve into the details of registration of elliptical patches for comparison. In our specific setting there are two approaches to registration:

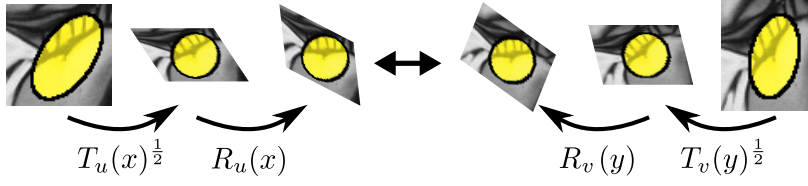


Figure 6.4: Registration of two shape-adaptive patches for comparison by normalization and additional rotation.

either one of the elliptical patches is transformed using the a priori connection (5.16), or both elliptical patches are normalized to circular patches as shown in Figure 6.4. The first approach will be exploited later on for image synthesis in application to image inpainting and denoising. Meanwhile the second option is more suitable for patch comparison, because it allows intermediate data caching. We first describe normalization and interpolation for 2D images and in the end of the section briefly comment on the differences in the \mathbb{R}^3 case.

In order to normalize an elliptical patch we need a proper transformation. As was shown previously, an initial transformation is obtained directly from a corresponding structure tensor. This preliminary normalization is used to estimate dominant orientations within the normalized patch. Several dominant orientation might be estimated for every patch; therefore, there can be several options for the normalizing transformations. Let us denote them here by $A_k(x) = R(\Theta_k)T_u(x)^{\frac{1}{2}}$, where $R(\Theta_k)$ is the additional rotation given by (6.8). Obviously, all the options are equivalent and we should consider each of them.

Since in practice digital images are discrete, after normalization any patch turns into a set of scattered points, example of which is shown in blue in Figure 6.5. In order to compare one normalized patch with another, we interpolate these sets of scattered points to a regular grid \mathcal{G} (shown in red in Figure 6.5). We consider the resolution of the regular grid to be constant throughout a single run of any experiment; thus, it can be built in the preprocessing phase. In terms of data structure the grid is represented by a set of real-valued coordinates of its nodes. Any normalized patch is contained within a circle of radius r ; therefore, the regular grid is built in such a way that its nodes evenly cover this circle with a given resolution. It is convenient to specify the resolution of the grid by the number of nodes that should fit along the diameter of the circle. We denote this free parameter

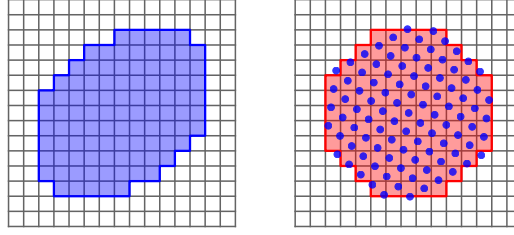


Figure 6.5: Elliptical patch normalization. On the left an elliptical patch is shown in blue. On the right a set of scattered points of that patch, being normalized by $R(\Theta_k)T_u(x)^{\frac{1}{2}}$ to a disc, is shown in blue over a regular grid which is shown in red. In this case, the grid resolution (number of grid nodes along the diameter) $g = 11$.

of the method by g . Algorithm 3 illustrates the construction of the regular grid.

In principle, after normalization we could interpolate one set of scattered points directly to another; however, the usage of the intermediate regular grid allows us to precompute all normalized patches and store them in memory. If we store all possible normalized versions for every elliptical patch, together with their corresponding candidate transformations $A_k(x)$, the calculation of the patch distance between two points boils down to several sums of squared differences as in (1.1).

For interpolation we use the Nadaraya-Watson estimator [Nadaraya \(1964\)](#); [Watson \(1964\)](#), with Gaussian kernel which standard deviation σ_G depends on the density of the scattered points. More specifically, it depends on the radius of a normalized patch (which is determined by the parameter r of the method) and on the number of points within a patch being normalized:

$$\sigma_G = \frac{r}{\sqrt{|B_u(x)|}}. \quad (6.10)$$

Algorithms 4 and 5 describe the normalization and interpolation of elliptical patches in a more formal way.

In the \mathbb{R}^3 case, ellipsoidal patches are normalized to spheres of radius r ; therefore, the regular grid is three-dimensional with its resolution equal in all dimensions. The normalization procedure is almost the same, with the difference that the additional rotations $R(\Theta_k)$ are now computed by (6.9). The interpolation is done using the same estimator with 3D Gaussian kernel

which standard deviation is

$$\sigma_{\mathcal{G}} = \frac{r}{\sqrt[3]{|B_u(x)|}}. \quad (6.11)$$

Algorithm 3: Construction of a regular grid in \mathbb{R}^2 .

Parameters: g, r

Output: \mathcal{G}

```

 $s \leftarrow \frac{2r}{g}$  // grid step
 $\mathcal{G} \leftarrow \{\}$  // set of grid nodes
for  $0 \leq i < g$  do
    for  $0 \leq j < g$  do
         $\omega \leftarrow [j s + 0.5 s - r, i s + 0.5 s - r]^t$  //  $\omega \in \mathbb{R}^2$ 
        if  $\|\omega\|_2^2 \leq r^2$  then
             $\mathcal{G} \leftarrow \mathcal{G} \cup \omega$ 

```

Algorithm 4: Normalization of elliptical patches.

Input: $u, T_u(x), B_{T_u}(x), x \in \mathbb{R}^2, \Theta_k$

Output: $\bar{B}_k(x)$

```

 $R \leftarrow \begin{bmatrix} \cos \Theta_k & \sin \Theta_k \\ -\sin \Theta_k & \cos \Theta_k \end{bmatrix}$  // additional rotation
 $A \leftarrow R T_u(x)^{\frac{1}{2}}$ 
 $\bar{B}_k(x) \leftarrow \{\}$ 
foreach  $y \in B_{T_u}(x)$  do
     $\bar{y} \leftarrow A(y - x)$ 
     $\bar{B}_k(x) \leftarrow \bar{B}_k(x) \cup (y, \bar{y}, u(y))$ 

```

Algorithm 5: Interpolation of normalized patches to a regular grid.

Input: $\bar{B}_k(x)$, \mathcal{G} , $\sigma_{\mathcal{G}}$
Output: $\bar{C} := \{\bar{c}_i\}_k$

```

 $\sigma := \sigma_{\mathcal{G}}$  // alias
 $\bar{C} \leftarrow \{\}$ 
foreach  $\omega \in \mathcal{G}$  do
     $\bar{c} \leftarrow 0$  // interpolated color value
     $W \leftarrow 0$  // total weight
    foreach  $(z, \bar{z}, c) \in \bar{B}_k(x)$  do
         $d \leftarrow \|\omega - \bar{z}\|^2$ 
         $w \leftarrow \exp(-\frac{d}{2\sigma^2})$  // current weight
         $\bar{c} \leftarrow \bar{c} + c w$ 
         $W \leftarrow W + w$ 
     $\bar{C} \leftarrow \bar{C} \cup \bar{c} W^{-1}$ 

```

Experimental Results for the Similarity Measures

In this chapter we illustrate the properties of the similarity measures (4.16) and (4.17), equipped with the affine covariant structure tensors. The first set of experiments demonstrates the multiscale property and the second set is devoted to the affine invariance property.

Throughout this work we refer to (4.16) and (4.17) as similarity measures; however, what we actually compute by $\mathcal{D}^a(t, x, y)$ (or, equivalently, by $C(t, x, y)$) is a *distance* (or dissimilarity) between two patches at points x and y , and scale t . We use the terms *similarity* and *distance* interchangeably, because in the context of image comparison they represent essentially the same relation between two patches in the whole space of patches.

There are two main parameters involved in the proposed affine invariant patch similarity measures that control the amount of support we allow in the comparison of two patches: t and r . In several experiment in this work both t and r are variable and, therefore, it is useful to link them together by

$$t = \left(\frac{r}{\hat{t}} \right)^2.$$

Small values of \hat{t} correspond to coarse scales (larger window), while big values correspond to fine scales (smaller window). We will call both t and \hat{t} scale parameters or simply “scales”. Notice, however, that t can be seen as an absolute scale, while \hat{t} as a relative scale with respect to r . Recall that in (4.17) g_t is a window that we assume to be Gaussian of variance

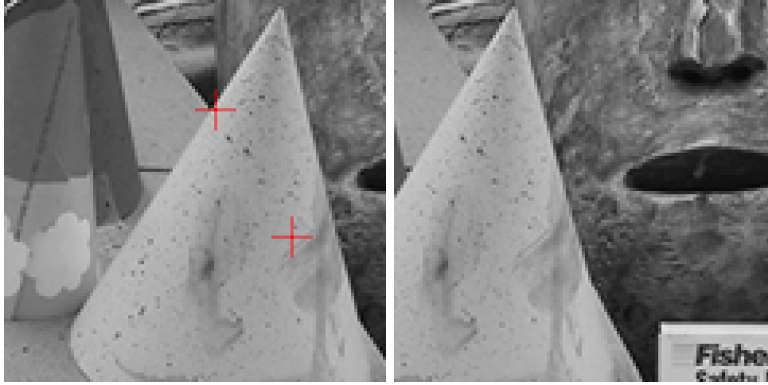


Figure 7.1: Stereo pair used in the first experiment. Two red crosses show the selected points of interest.

t . Then informally \hat{t} can be seen as the number of Gaussian sigmas that should fit into a patch, which in the \mathbb{R}^2 case has elliptical shape and can be normalized to a circle of radius r . By introducing \hat{t} we can specify one set of scales for all values of r .

7.1 Multiscale Property

In this section we concentrate on the multiscale property of the proposed patch similarity measures. In many cases the scale of patch comparison can be selected once and for all; however, it might also be helpful to compare patches at several scales simultaneously. The following experiments are aimed to demonstrate the behavior of the similarity measures at different scales and motivate the usage of multiple scales. For illustrative purposes we select the problem of disparity map estimation. Let us remark that even though all the experiments in this section were made in the context of disparity map estimation, the complete application of the proposed similarity measure to depth estimation from stereo images is out of the scope of the current work.

First Experiment

In the first experiment we compute similarity values between a given point in the left image and all the points in the right image of the stereo pair shown in Figure 7.1. One point was selected in the interior of an object in

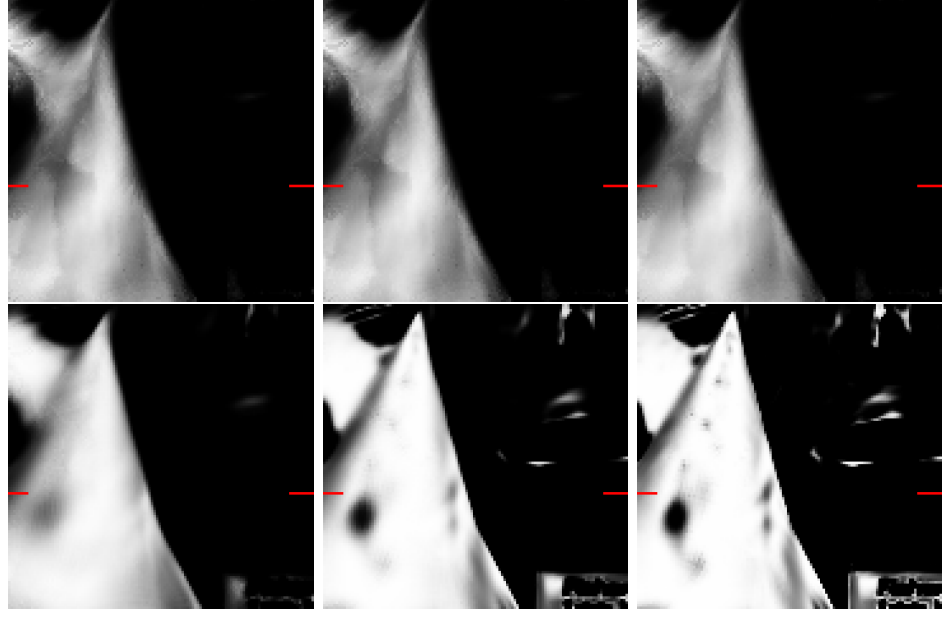


Figure 7.2: Similarity maps for the point inside the object calculated using the similarity measure (4.17) with Gaussian weights. From left to right and top to bottom: \hat{t} is set to 0.01, 0.5, 1, 3, 6 and 9.

the left image and another point on a boundary of an object. In the figure these points are marked with red crosses.

In this experiment the radius was set to $r = 150$ and the scale parameter was selected to be $\hat{t} \in \{0.01, 0.5, 1, 3, 6, 9\}$. A small value of \hat{t} (e.g. $\hat{t} = 0.01$) corresponds to a coarse scale (large window), while a big value (e.g. $\hat{t} = 9$) corresponds to a fine scale (small window).

For visualization purposes we color-code the patch distances by

$$c(x, y, t) = 255 \exp \left(-\frac{(\mathcal{D}^a(t, x, y) - \mathcal{D}_{min}^a)^2}{2\sigma^2} \right), \quad (7.1)$$

where $\sigma = \frac{\mathcal{D}_{max}^a - \mathcal{D}_{min}^a}{\gamma}$, \mathcal{D}_{max}^a and \mathcal{D}_{min}^a are the maximum and minimum patch distance values, respectively, and $\gamma > 0$ is a visualization parameter. Notice that the highest value of $c(x, y, t)$ corresponds to the most similar patches. For all experiments in this section we set $\gamma = 50$, which stretches values close to 1. The values of \mathcal{D}_{max}^a and \mathcal{D}_{min}^a were computed from the whole set of distances across all the scales, which means that equal colors on two different similarity maps correspond to the same similarity values.

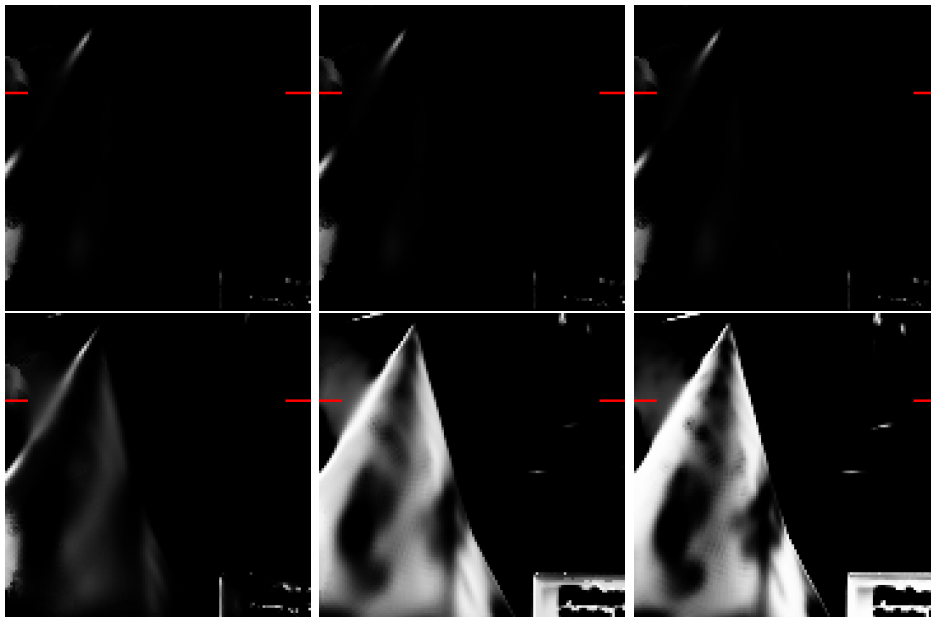


Figure 7.3: Similarity maps for the point on the boundary calculated using the similarity measure (4.17) with Gaussian weights. From left to right and top to bottom: \hat{t} is set to 0.01, 0.5, 1, 3, 6 and 9.

Figures 7.2 and 7.3 show similarity maps computed using the similarity measure (4.17). Figure 7.2 shows maps for the point selected in the interior of the object, while Figure 7.3 shows maps for the point selected on the boundary. Similarly, Figures 7.4 and 7.5 show similarity maps for the same setting, but computed with approximated geodesic weights (4.16) as in the bilateral filter (with $\kappa_{spatial} = 1.0$ and $\kappa_{color} = 3.0$). Let us note that in the case of disparity map estimation from stereo images it is common to restrict the search for the most similar point to the epipolar lines or, when the images are rectified, to the corresponding scan lines. The scan lines in which we would be interested while estimating disparities are marked with small red strokes in the figures.

Discussion

The first point was selected inside an object on a region with a smooth texture, far away from the boundary of that object. In this case, one would expect that the better matching can be achieved with a larger window. It is confirmed by the experiments (Figures 7.2 and 7.4), showing that small

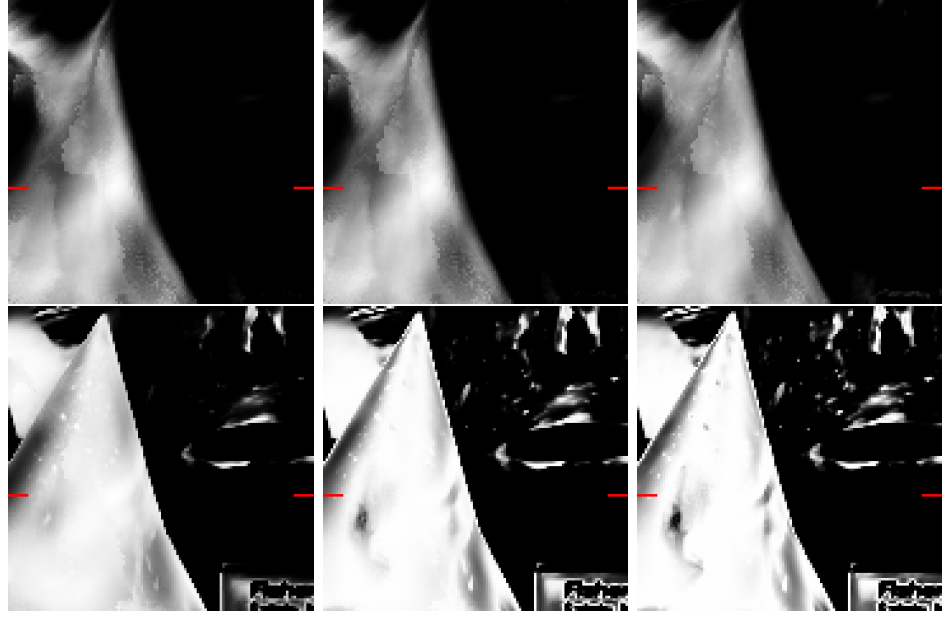


Figure 7.4: Similarity maps for the point inside the object calculated using the similarity measure (4.16) with geodesic weights. From left to right and top to bottom: \hat{t} is set to 0.01, 0.5, 1, 3, 6 and 9.

values of \hat{t} give more distinctive matching and better localization, whereas big values (for example, $\hat{t} = 9$) produce equally high similarity values for many adjacent points.

The point on a boundary of the object was selected in such a way that the background of that object undergoes a severe change (Figures 7.3 and 7.5). This determines very low similarities for small values of \hat{t} when the influence of the background is high due to the large window. The difference is especially noticeable with the geodesic weights which produce sharp edges in the similarity maps.

The underlying premise of geodesic weights is that colors or intensities can be used to distinguish between different objects. That is, the pixels in the neighborhood of x with a color similar to $u(x)$ have more influence on the similarity calculation than pixels with different colors, which probably belong to another object.

Note that in the approximated geodesic distance in (4.16) the influence of the image colors is controlled by κ_{color} . The geodesic weights on a region

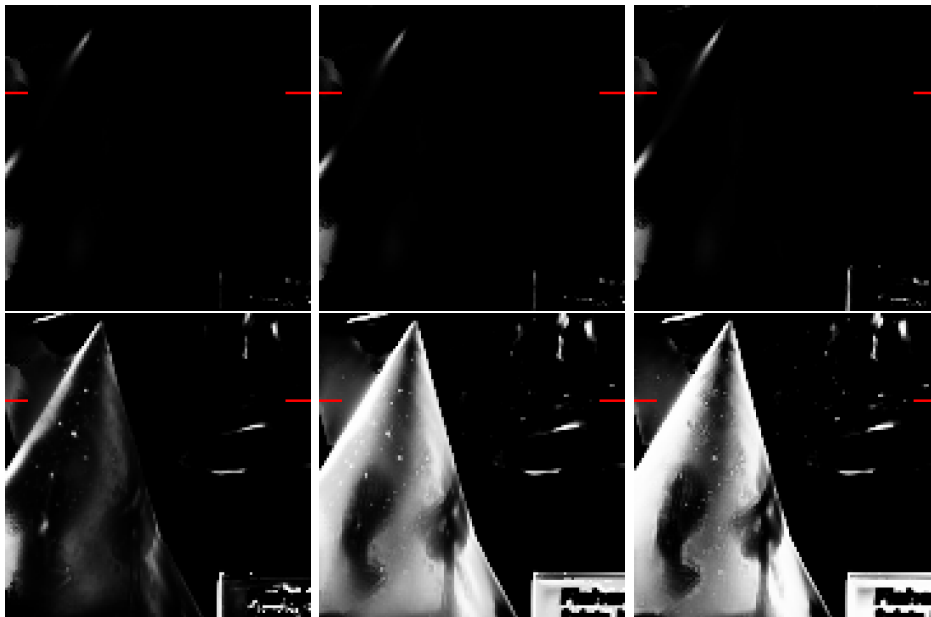


Figure 7.5: Similarity maps for the point on the boundary calculated using the similarity measure (4.16) with geodesic weights. From left to right and top to bottom: \hat{t} is set to 0.01, 0.5, 1, 3, 6 and 9.

with similar color are virtually identical to the Gaussian weights. However, if the window contains two objects with different colors, then (for a reasonable choice of κ_{color}) only the pixels with a color similar to the center of the window have a significant contribution in the similarity calculation. Finally, by taking a very large value for κ_{color} one would recover the Gaussian weights (4.17) regardless of the image colors. The choice of κ_{color} is delicate as it should be small to distinguish objects by their intensity, but large enough to capture the variations of intensity within an object. In the context of a denoising application the value of this parameter is usually chosen to be proportional to the noise present in the image (Tomasi and Manduchi (1998)).

In stereo-vision the geodesic weights are particularly important, because matching using fixed weights produces the so-called *foreground fattening* effect. This phenomenon occurs when a matching window contains parts of objects with different depths. In this setting, background pixels near an occluding edge may get the depth of the occluding edge (which is in the foreground), hence in the estimated depth map the foreground object



Figure 7.6: Stereo pair used for disparity maps calculation.

appears fattened.

The geodesic weights correspond to a well-established stereo-vision technique of Yoon and Kweon (2006) that allows to estimate sharper depth maps near depth discontinuities using weights similar to the bilateral filter of Tomasi and Manduchi (1998).

It is worth mentioning, that for the small values of \hat{t} , points close to the tip of the cone (Figures 7.3 and 7.5) have much higher similarity values, than points on the scan line. This happens because in these locations background has more similar appearance to the one at the reference point. In applications other than stereo this might be a desirable best match.

Second Experiment

In order to show the effect of changing the scale on the whole image, we calculate disparity maps for the stereo pair shown in the Figure 7.6. For every point in the left image we compute the disparity by exhaustive search for the best match in the range of possible offsets $[-55, -5]$ along the scan line in the second image. For this experiment the r parameter is fixed as before to $r = 150$ and the scale parameter varies in $\hat{t} \in \{0.01, 1, 3, 6, 9\}$. For approximated geodesic weights (equation (4.16)), the coefficients were set to $\kappa_{spatial} = 1.0$ and $\kappa_{color} = 3.0$.

Figure 7.7 shows the ground truth disparity map and the disparity maps computed at different scales using the similarity measure (4.17). Figure 7.8 shows the occlusion mask and the errors with respect to the ground truth excluding the occlusion mask. Similarly, Figures 7.9 and 7.10 show

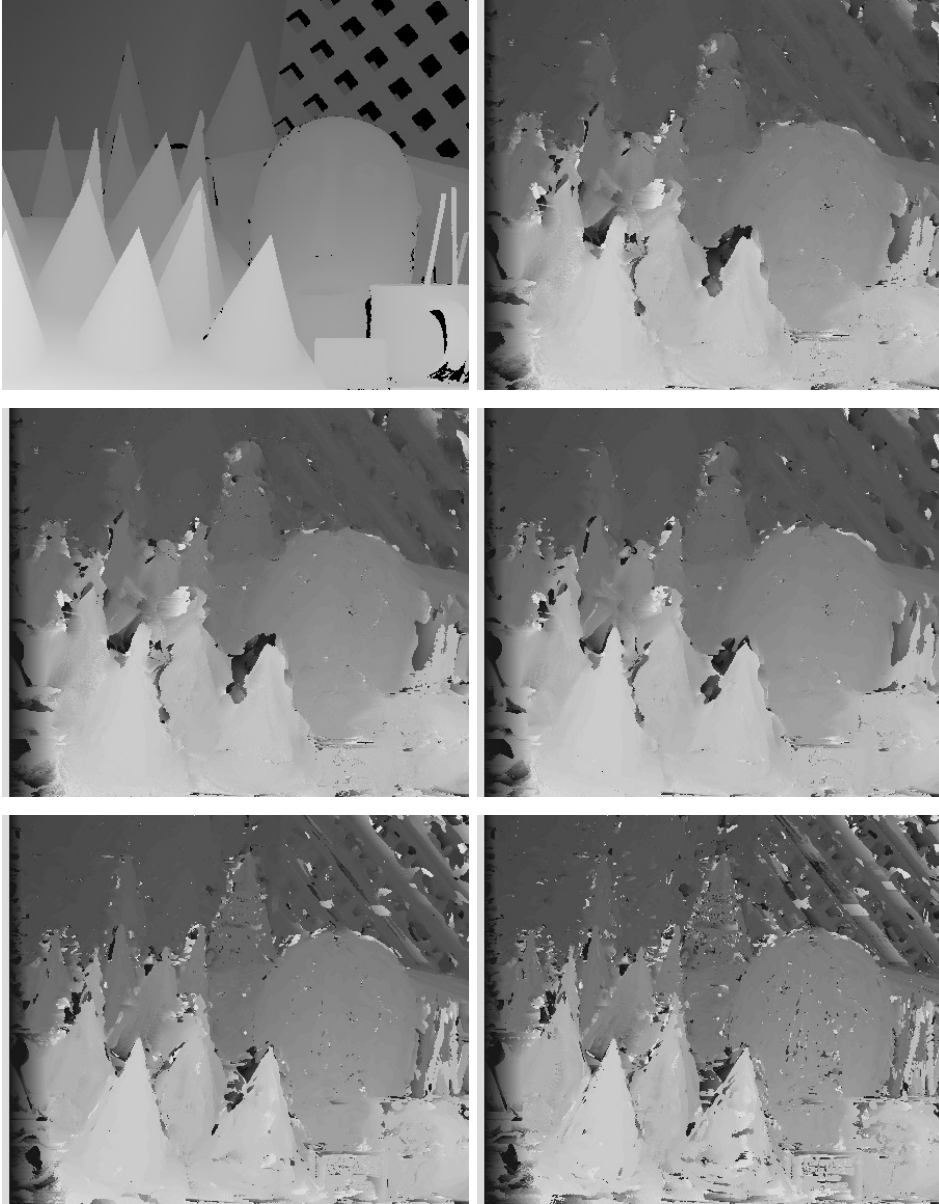


Figure 7.7: Disparity maps calculated using the similarity measure (4.17) with Gaussian weights. Brighter colors correspond to bigger offsets. From left to right and top to bottom: ground truth, disparity maps for \hat{t} set to 0.01, 1, 3, 6, 9.

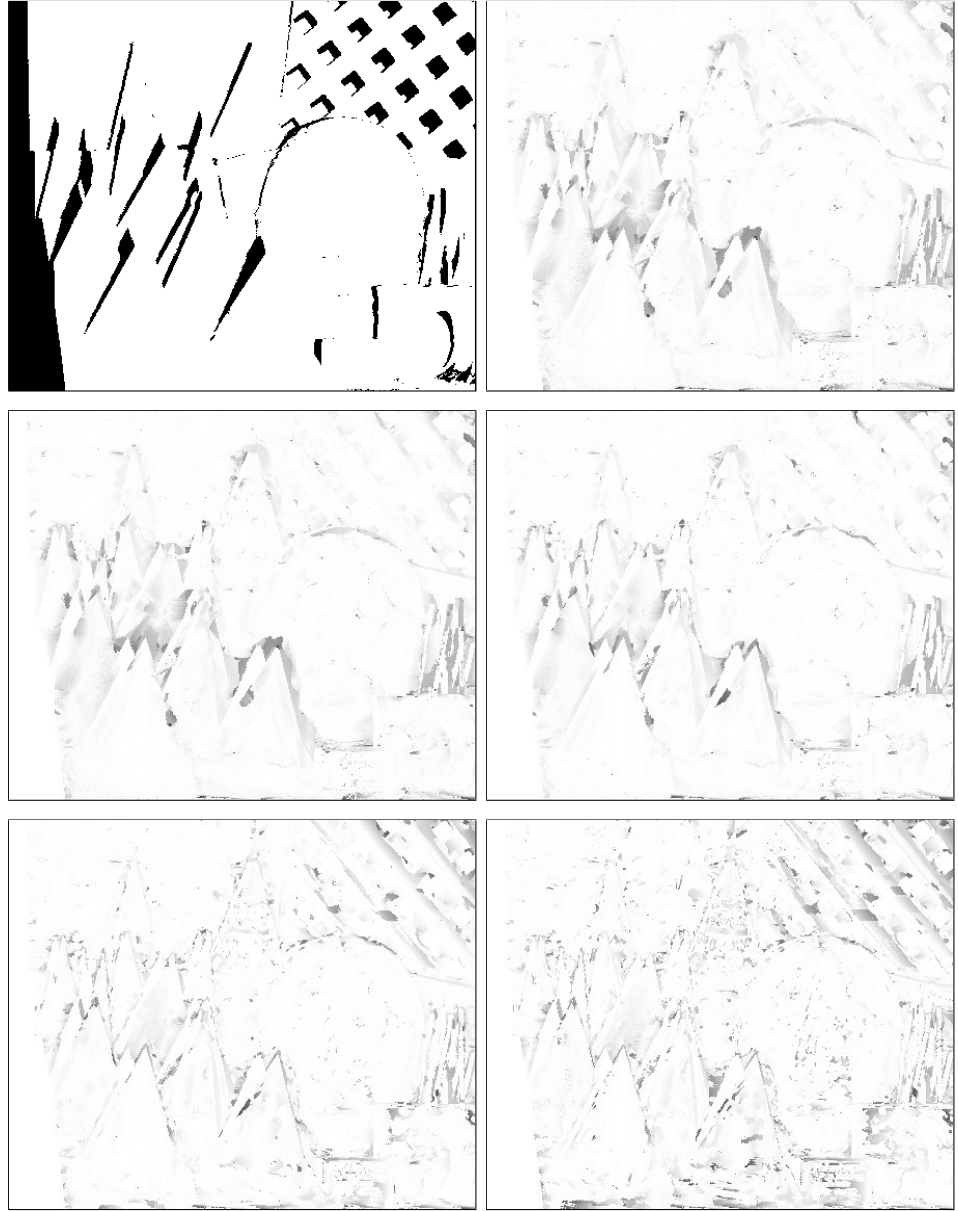


Figure 7.8: Errors in disparities with respect to the ground truth excluding the occlusion mask. Disparities were calculated using the similarity measure (4.17) with Gaussian weights. Darker colors correspond to bigger errors. From left to right and top to bottom: occlusion mask, error maps for \hat{t} set to 0.01, 1, 3, 6, 9.

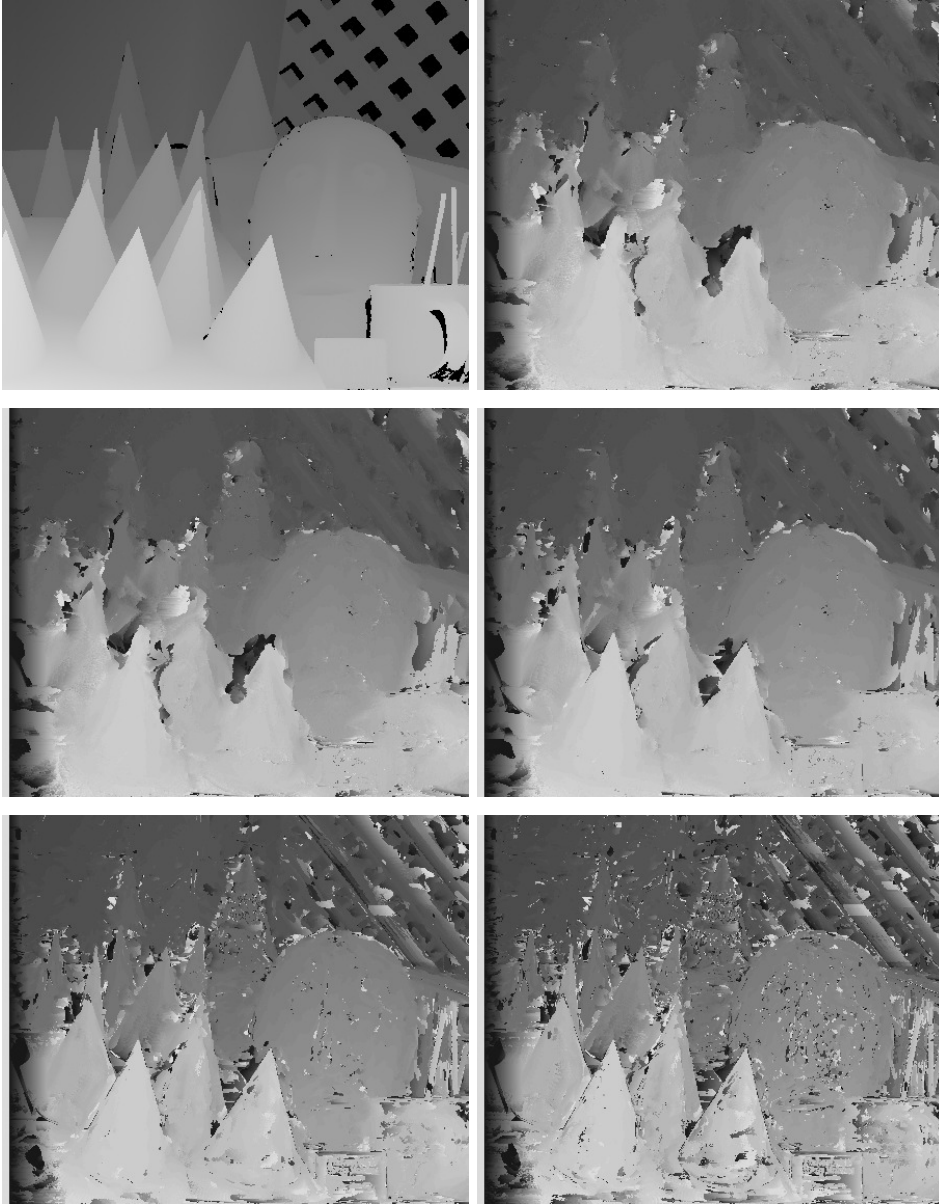


Figure 7.9: Disparity maps calculated using the similarity measure (4.16) with geodesic weights. Brighter colors correspond to bigger offsets. From left to right and top to bottom: ground truth, disparity maps for \hat{t} set to 0.01, 1, 3, 6, 9.

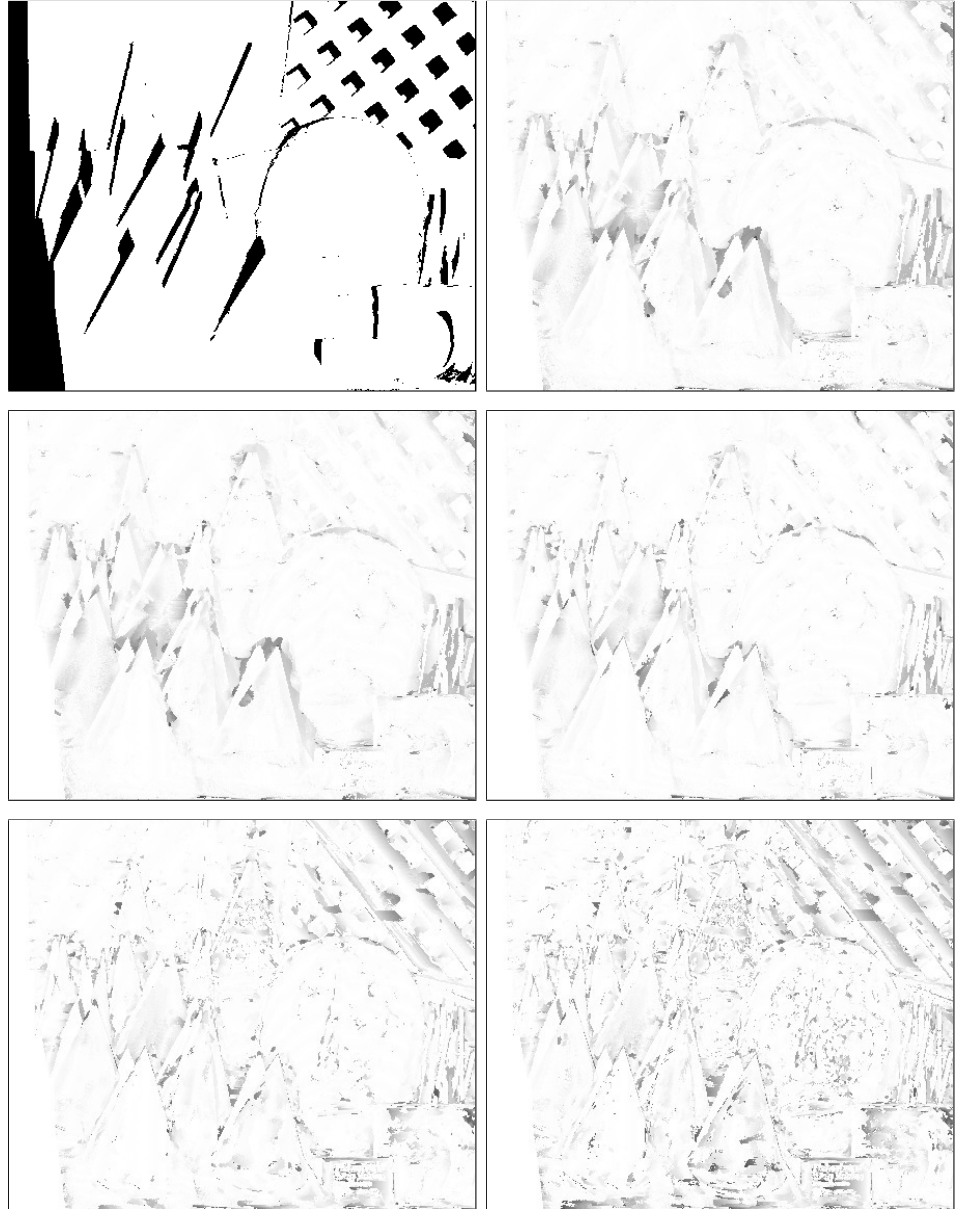


Figure 7.10: Errors in disparities with respect to the ground truth excluding the occlusion mask. Disparities were calculated using the similarity measure (4.16) with geodesic weights. Darker colors correspond to bigger errors. From left to right and top to bottom: occlusion mask, error maps for \hat{t} set to 0.01, 1, 3, 6, 9.

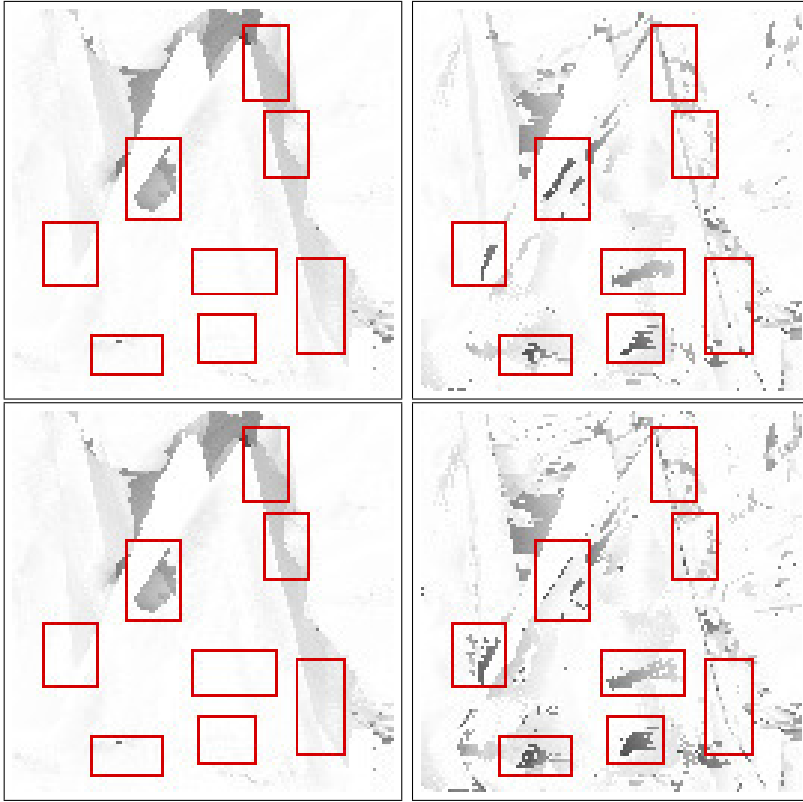


Figure 7.11: Closed up errors with respect to the ground truth excluding the occlusion mask. First row: calculated using (4.17) for $\hat{t} = 0.01$, $\hat{t} = 9$. Second row: calculated using (4.16) for $\hat{t} = 0.01$, $\hat{t} = 9$.

disparity maps and errors computed using the similarity measure (4.16) with approximated geodesic weights.

Discussion

As it can be seen, calculating similarity values on a coarse scale results in smooth disparity maps which closely match the ground truth within the objects but are over-smoothed at the boundaries. In contrast, fine scale similarities emphasize sharp boundaries of objects, but produce a lot of mismatches inside flat regions.

Figure 7.11 shows the closed up fragments of error maps for the limit cases ($\hat{t} = 0.01$ and $\hat{t} = 9$). Some of the interesting regions are highlighted in red.

This illustrates the point that in general, if a foreground object is moving over its background, the similarity measure performs differently at different scales within the object and close to its boundary.

In conclusion, in the context of stereo-imaging and for some other applications it is desirable to look for the closest patch across multiple scales. For that, patches can be compared at several scales simultaneously and then multiple responses can be “merged” in one way or another. The exact approach to the multiscale patch comparison may vary depending on the application. In the ones described in Part II we select a single scale for patch comparison beforehand. The issue of combining responses from different scales is out of the scope of the current work and should be studied in detail in future.

7.2 Affine Invariance Property

In this section we focus on the affine invariance property of the proposed patch similarity measures (4.16) and (4.17).

Similarity Measures under Different Transformations

At first we evaluate performance of the proposed affine invariant similarity measures on pairs of images, which undergo different kinds of transformations. For this purpose we calculate patch distance values $\mathcal{D}^a(x, y, t)$ between corresponding points in two images. In each selected pair of images, differing by an affine or planar projective transform, the ground truth correspondences are either given (Figure 7.12), or easy to estimate (Figures 7.14 and 7.16). In the following experiments we demonstrate results for a rotation, an affine transformation and finally for a perspective transformation. In each case we compare the proposed affine invariant similarity measure (4.17) with the well-known Euclidean distance given by (1.1). Let us remark that (1.1) can be seen as a particular case of (4.17), where $T_u(x) = I$ and $T_v(y) = I$. This similarity measure is modeled by a usual patch distance between square patches of a given size. The results for the similarity measure (4.16) with geodesic weights are very alike; therefore, we do not show them.

For both patch similarity measures being compared we test several sizes of patches. For (1.1) we directly set the size of square patches in pixels in the range of $[5, 23]$. For the proposed similarity measure (4.17) we set the r parameter in the range of $[50, 300]$. In the described setting there is only one

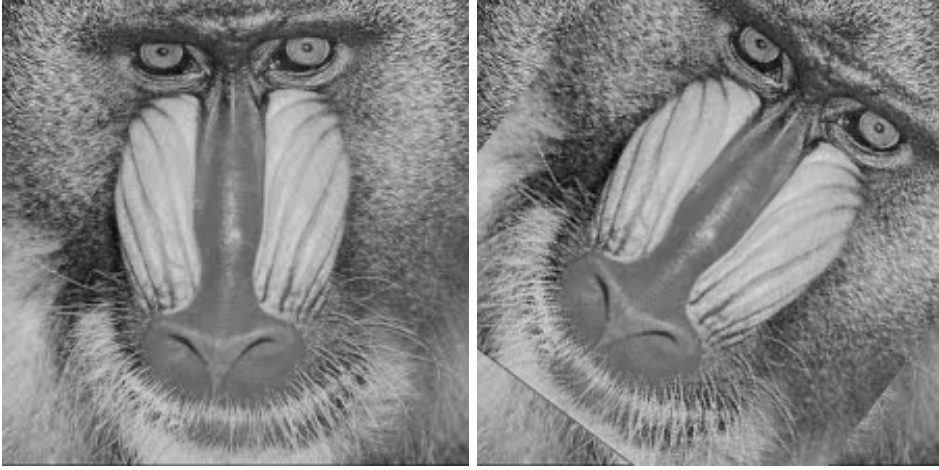


Figure 7.12: Original image and rotation of the original image by an angle of 37 degrees superimposed on the original.

object shown in a pair of images under different transformations; moreover, the ground truth correspondences are provided. Therefore, for the following experiments we fix the scale parameter $\hat{t} = 0.01$, which correspond to almost uniform weights within the shape-adaptive patches.

As before, in order to visualize similarity maps, we color-code patch distances $\mathcal{D}^a(x, y, t)$ by (7.1). Recall that $\gamma > 0$ is a visualization parameter that controls stretching of values close to 1. Brighter color means higher similarity value. For each pair of images, distances were normalized all together using the same \mathcal{D}_{min}^a and \mathcal{D}_{max}^a , which means that equal colors on two different similarity maps correspond to the same similarity values.

First experimental setting. In the first experiment we verify the invariance to rotations on a pair of images shown in the Figure 7.12. The left image is the original one, and the right image was rotated clockwise by an angle of 37 degrees. Figure 7.13 shows similarity values calculated from the given correspondences. The distances were color-coded with $\gamma = 10$.

Second experimental setting. For the second experiment we use a pair of images (Figure 7.14) related by an affine transform. The left image is the fronto-parallel view and the right image is the affinely-rectified view of the same scene. Figure 7.15 shows similarity values calculated from the given correspondences. The distances were color-coded with $\gamma = 20$.

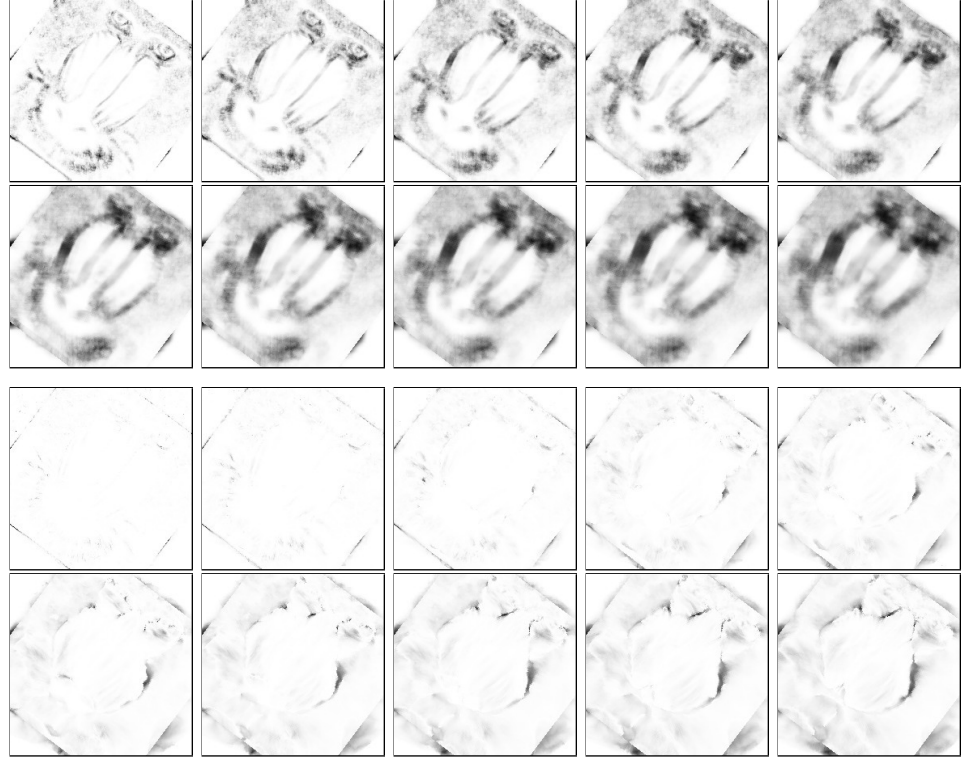


Figure 7.13: Similarity maps for given correspondences between the original and rotated images. First and second rows: similarities calculated using square patches of sizes 5, 7, 9, 11, 13; 15, 17, 19, 21, 23. Third and fourth rows: similarities calculated using the proposed similarity measure (4.17) with r set to 50, 75, 100, 150, 175; 200, 225, 250, 275, 300.

Third experimental setting. Third pair of images (Figure 7.16) includes two different views of the same packet of juice. We are interested in the front side of the packet, which is planar. Two given views of that front side are related by perspective transform of a plane. Figure 7.17 shows similarity values calculated from the given correspondences. The distances were color-coded with $\gamma = 10$.

As expected, in all three settings and for the whole ranges of square patch sizes and values of r , the proposed affine invariant similarity measure outperforms the usual Euclidean distance (1.1). Since a projective transformation can locally be approximated by an affinity, the proposed similarity measure was able to handle the third case. These results show also that the param-



Figure 7.14: Fronto-parallel and affinely-rectified views of the same scene.

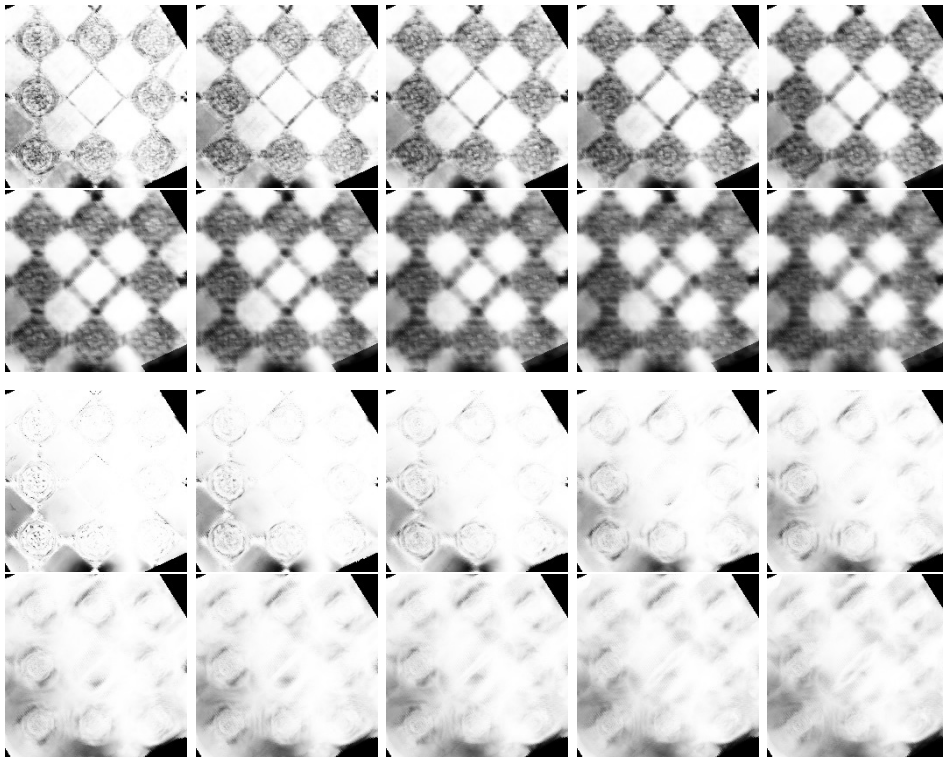


Figure 7.15: Similarity maps for given correspondences between the images related by an affinity. First and second rows: similarities calculated using square patches of sizes 5, 7, 9, 11, 13, 15, 17, 19, 21, 23. Third and fourth rows: similarities calculated using the proposed similarity measure (4.17) with r set to 50, 75, 100, 150, 175, 200, 225, 250, 275, 300.

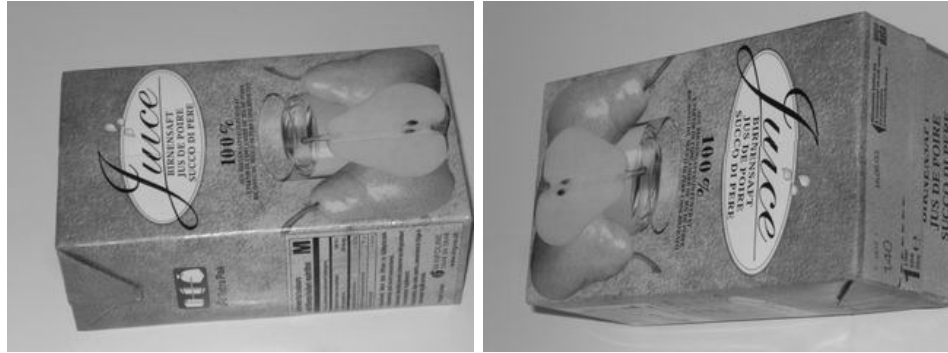


Figure 7.16: Two views of the same packet of juice.

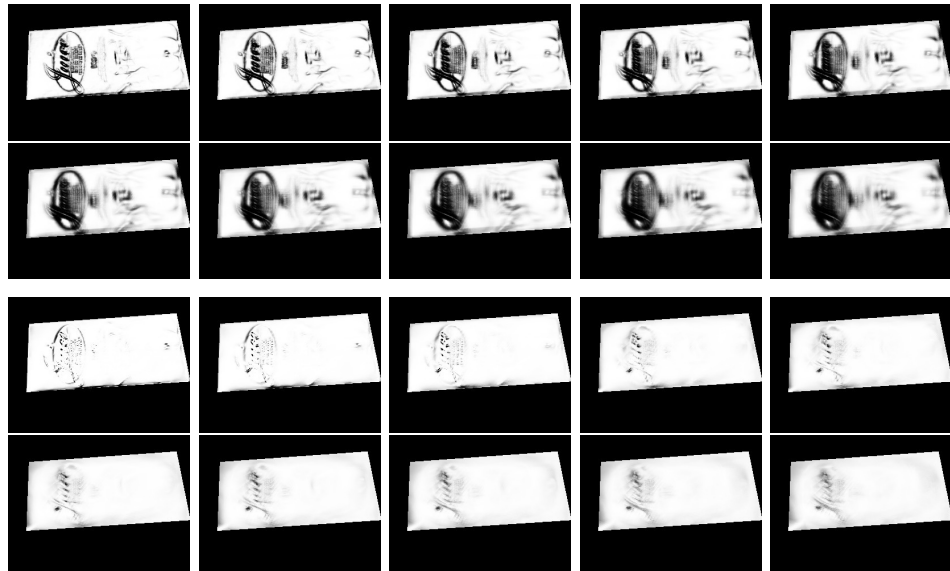


Figure 7.17: Similarity maps for given correspondences between two views of the front side of the packet of juice. First and second rows: similarities calculated using square patches of sizes 5, 7, 9, 11, 13; 15, 17, 19, 21, 23. Third and fourth rows: similarities calculated using the proposed similarity measure (4.17) with r set to 50, 75, 100, 150, 175; 200, 225, 250, 275, 300.



Figure 7.18: Pair of images used to study the robustness of the proposed affine invariant similarity measures in regions close to the boundary of two different transformations. The region of interest is marked by a red dashed square.

eter r may vary in a relatively wide range without significant impact on the resulting similarity values. The actual choice of r should depend on the application, the sizes of the image features to be captured, the image noise and texture content itself.

Conjunction of Transformations

Finally, we study the behavior of the affine invariant similarity measures (4.16) and (4.17) in the discrete setting for regions close to a boundary between two different transformations applied to parts of the digital image. For this we created a pair of synthetic images, shown in Figure 7.18. In these images two different views from the *graffiti* sequence are used as backgrounds and two different views of another object are imposed as foreground objects. Therefore, the backgrounds are related by one transformation while the foreground objects are related by another transformation. The ground truth correspondences are calculated first for the backgrounds, then for the foreground objects and finally combined using a mask.

Figure 7.19 shows similarity maps calculated from the known correspondences for different values of r and scale parameter \hat{t} . The set of r values is $r \in \{50, 100, 150, 200\}$ and the set of \hat{t} values is $\hat{t} \in \{0.01, 3, 9\}$. Notice that the similarity maps are computed only for a region of interest around the foreground object. This region is shown as a red dashed square in Figure 7.18. The similarities are computed using both similarity measures (4.16) and (4.17). In the later case the geodesic distance is approximated as in the bilateral filter of Tomasi and Manduchi (1998) with $\kappa_{spatial} = 1.0$ and $\kappa_{color} = 3.0$. For visualization purposes the similarity values are linearly mapped into intensity values. Brighter color corresponds to higher

similarity.

The similarity maps in figure 7.19 show that for all selected combinations of \hat{t} and r the matching is correct even close to the boundary of different transformations with some expected errors right at the boundary. The problematic region at the boundary is smaller for the similarity measure (4.16) with geodesic weights which naturally separates objects with different colors. In general, accuracy at the boundary between two different transformations can be controlled by the scale parameter \hat{t} . This emphasizes once again the importance of a multiscale similarity computation for the cases where two or more objects, undergoing different transformations, meet each other.

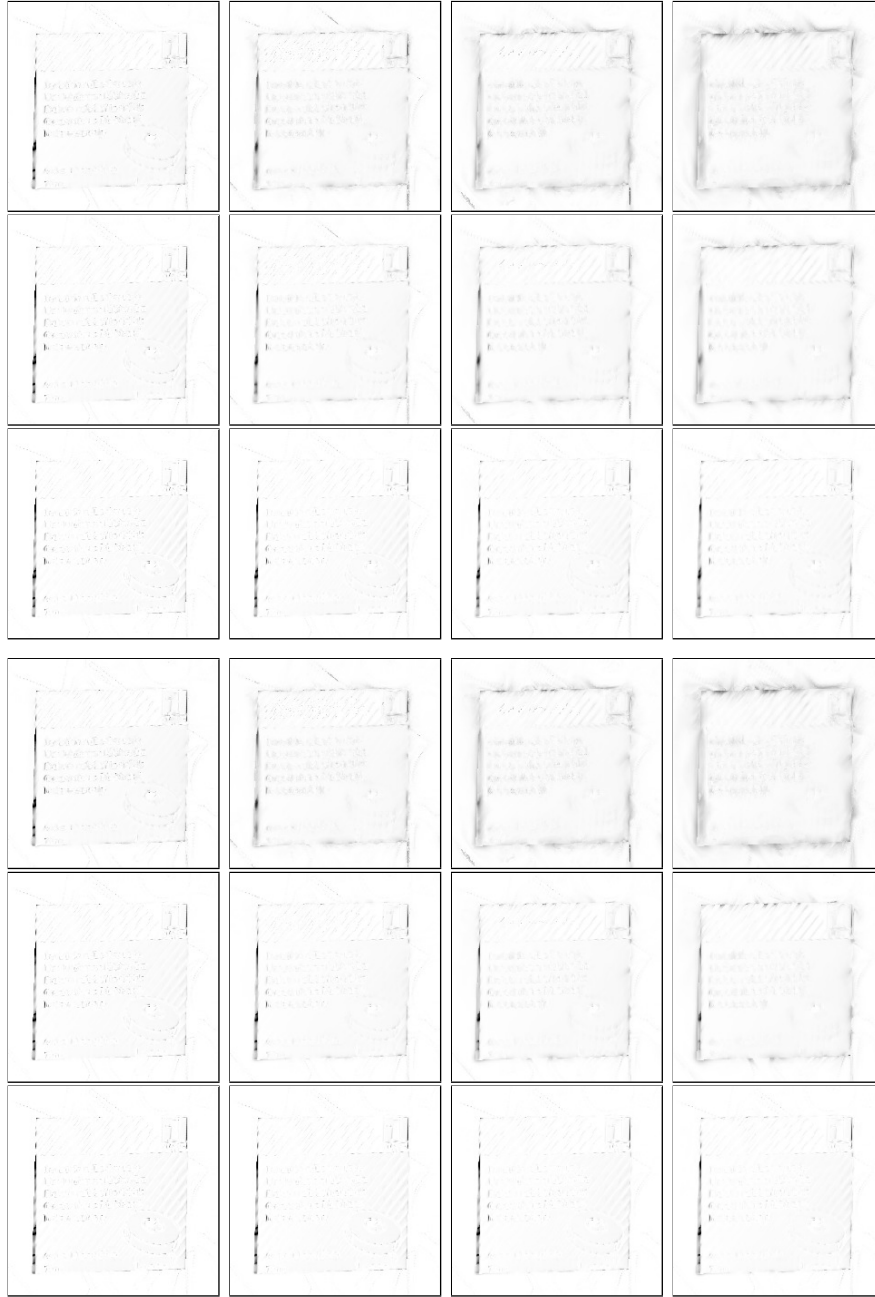


Figure 7.19: Similarity values for given correspondences, calculated for different values of \hat{t} and r . Columns: r is equal to 50, 100, 150 and 200. First three rows: Gaussian weights (4.17), \hat{t} is equal to 0.01, 3 and 9. Last three rows: geodesic weights (4.16) with $\kappa_{\text{spatial}} = 1.0$ and $\kappa_{\text{color}} = 3.0$, \hat{t} is equal to 0.01, 3 and 9.

Conclusions

In the first part of the thesis we have studied the linear model of affine invariant patch-wise image comparison for images defined on Riemannian manifolds. This generic and at the same time practical linear model was originally described in [Ballester et al. \(2014\)](#) within a broad framework of multiscale analyses of similarity measures. Throughout this work, speaking about image comparison, we have in mind a patch similarity measure assigning a similarity (distance) value to a given pair of points at two images.

In an attempt to keep the thesis self-contained, we have collected the necessary notation and have given an overview of the underlying mathematical concepts, related to Riemannian manifolds. We have also presented the axiomatic approach used in [Ballester et al. \(2014\)](#) to classify and study multiscale analyses of similarity measures, in an amount that is sufficient for derivation of the linear model we are interested in. Following the previous work, we have shown that multiple multiscale similarity measures can be obtained as solutions of partial differential equations corresponding to the linear model. Since in general the model involves spatially varying Riemannian metrics in image domains, a direct solution of such PDE would have computational complexity of order S^4 , if both images being compared are defined on a grid of size S^2 . Therefore, using WKB method, we have developed an approximate solution leading to a family of patch similarity measures that are well-suited for practical applications. This approximation, together with two particular multiscale similarity measures (4.16) and (4.17), should be considered as the first major contribution of the thesis.

After deriving two multiscale similarity measures, we have focused on the problem of defining the Riemannian metrics, such that the similarity mea-

asures are affine invariant. Our second contribution is the novel definition and the iterative computational scheme for the structure tensors, that can be used as metrics in image domains. We have shown that the structure tensors, defined in the proposed way, are guaranteed to be affine covariant in the contentious setting of infinite resolution images. We have also discussed the limitations, implied by the discrete setting and real images.

Although the formalism of image comparison studied in this work holds for images defined in \mathbb{R}^N , the most common and thus practical cases are obviously \mathbb{R}^2 and \mathbb{R}^3 . The latter covers, for example, 3D medical images and regular videos. We have discussed the most important details of the efficient numerical implementation of the proposed similarity measures in \mathbb{R}^2 and \mathbb{R}^3 . This should be considered as our next contribution.

Finally, we have experimentally studied the multiscale and affine invariance properties of the proposed similarity measures. We have shown that the multiscale image comparison might be beneficial when two or more objects within a scene undergo different transformations and occlude each other. For example, this is normally the case in the context of stereo-imaging. As expected, the proposed affine invariant similarity measures outperform the simple Sum of Squared Differences (1.1) in comparison of two images related by different transformations. A more rigorous study of the extent, to which the proposed similarity measures are affine invariant in practice, is a subject for future work. Another direction of the future research is a detailed comparison of both similarity measures in terms of performance and applicability in different contexts. Moreover, many other similarity measures covered by the linear model are left beyond the scope of this work. Even more possibilities offered by the axiomatic approach are yet to be explored.

In the second part of the thesis we present two particular applications of the similarity measure (4.17) to image inpainting and image denoising.

PART II

Applications

Introduction

The second part of the thesis is dedicated to applications of the affine invariant patch similarity measures. Our aim in this chapter is to recap the important components of the proposed patch similarity measures. Then in Chapter 10 we describe the application to exemplar-based image inpainting and in Chapter 11 we cover the extension of Non-Local Means image denoising method. We have decided to pursue these particular applications, because on the one hand inpainting results can be very much illustrative, and on the other hand denoising performance is easy to measure quantitatively. Finally, Chapter 12 concludes our work.

9.1 An Affine Invariant Similarity Measure

In Part I two examples of an affine invariant patch similarity measure were derived from the theoretical framework of multiscale analyses of similarity measures. In order to give a better intuition, in this section we look at them from a slightly different perspective. We begin with the affine covariance property of the proposed structure tensor and construct from it an affine invariant patch similarity measure.

Affine Covariant Structure Tensors

First of all we recall the notion of affine covariant structure tensors and the way they can be used to define shape-adaptive patches. Given a real-valued image $u : \Omega_u \rightarrow \mathbb{R}$, we consider an image-dependent structure tensor field T_u as a function that associates a structure tensor (a symmetric, positive semi-definite 2×2 matrix) with each point x in the image domain Ω_u . For

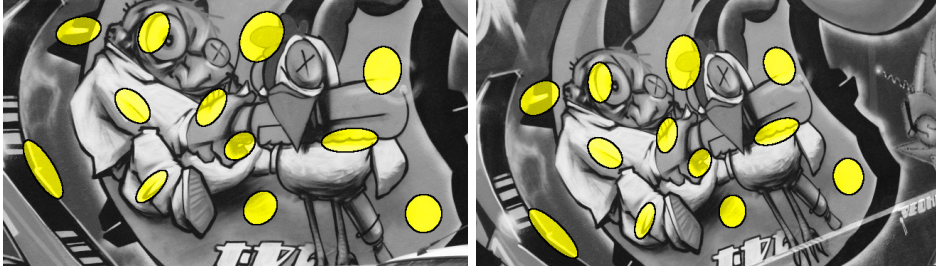


Figure 9.1: Affine covariant neighborhoods (shape-adaptive patches) computed at corresponding points in two images taken from different viewpoints. Despite the change in appearance, patches capture the same visual information.

simplicity we assume here the image domain to be \mathbb{R}^2 . The structure tensor field is said to be *affine covariant* if, for any affinity A ,

$$T_{u_A}(x) = A^T T_u(Ax) A, \quad (9.1)$$

where $u_A(x) := u(Ax)$ denotes the affinely transformed version of u . Given a structure tensor $T_u(x)$ we can associate with it an elliptical region of “radius” r centered at x

$$B_u(x, r) = \{y \in \mathbb{R}^2 : \langle T_u(x)(y - x), (y - x) \rangle \leq r^2\}. \quad (9.2)$$

When a structure tensor is affine covariant, we obtain an affine covariant elliptical region, such that

$$AB_{u_A}(x, r) = B_u(Ax, r).$$

This means that the structure tensors can be used to define affine covariant regions which transform properly via an affinity. We refer to these affine covariant regions as *shape-adaptive* or *elliptical patches* to distinguish them from the square patches of fixed size. Figure 9.1 demonstrates elliptical patches defined by affine covariant structure tensors, computed for a set of corresponding points in two images related by a homography. Recall that the efficient way to collect all points belonging to an elliptical patch was proposed in Section 6.2.

As shown in Section 5.2, given two affine covariant structure tensors we can extract the affine transformation between the corresponding elliptical patches up to some rotation. Indeed, for any affine transformation A , there exists an orthogonal matrix R such that

$$A = T_u(Ax)^{-\frac{1}{2}} R T_{u_A}(x)^{\frac{1}{2}}. \quad (9.3)$$

The equation above provides an intuitive geometric relationship between the structure tensors, the associated elliptical patches and the affinity. Consider a point x and the corresponding elliptical patch $B_{u_A}(x)$. Transforming $B_{u_A}(x)$ by an affinity yields $B_u(Ax)$. The application of affinity A can be decomposed in three steps. First, $T_{u_A}(x)^{\frac{1}{2}}$ transforms $B_{u_A}(x)$ into a disc or radius r . We refer to the resulting disc as a *normalized patch*. Then, a rotation R is applied to the normalized patch. Finally, $T_u(Ax)^{-\frac{1}{2}}$ maps the rotated normalized patch to the elliptical patch $B_u(Ax)$.

To fully determine the affinity A , one needs to find the rotation R . Any rotation would yield an affinity that maps the elliptical patch associated with T_{u_A} at x to the one associated with T_u at Ax . However, for a wrong value of the rotation the image content inside both patches will not match. Therefore, the correct value for the rotation can be computed by aligning the image content of both patches. For this aim, we decompose the rotation as

$$R = R_u^{-1}(Ax)R_{u_A}(x), \quad (9.4)$$

where $R_u(Ax)$ and $R_{u_A}(x)$ are estimated from the image content inside the patches. In practice, we calculate them by aligning dominant orientations of the normalized patches. To compute the dominant orientations we use histograms of gradient orientations as explained in Section 6.3.

Computation of Affine Covariant Tensors

Let us recall the particular definition and the iterative scheme proposed in Section 5.1 that allows us to compute a dense field of affine covariant tensors and associated neighborhoods:

$$T_u^{(k)}(x) = \frac{\int_{B_u^{(k-1)}(x,r)} Du(y) \otimes Du(y) dy}{\text{Area}(B_u^{(k-1)}(x,r))}, \quad (9.5)$$

where u is the given image and $B_u^{(k)}$ is the affine covariant neighborhood related to $T_u^{(k)}$, defined by

$$B_u^{(k)}(x,r) = \{y : \langle T_u^{(k)}(x)(y-x), (y-x) \rangle \leq r^2\} \quad (9.6)$$

for $k \geq 1$, and

$$B_u^{(0)}(x,r) = \{y : |Du(x)(y-x)| \leq r\} \quad (9.7)$$

for $k = 0$. As before, to simplify the notation we denote by $T_u(x)$ the affine covariant structure tensor $T_u^{(k)}(x)$ for a fixed values of k and r ($r > 0$

is a free parameter). Similarly, we denote by $B_u(x)$ the affine covariant neighborhood $B_u^{(k)}(x, r)$.

An Affine Invariant Patch Similarity

Previously in this section we were considering two images u and u_A , related by a global affinity. For the patch comparison problem we can generalize our reasoning and consider two arbitrary images $u : \Omega_u \rightarrow \mathbb{R}$ and $v : \Omega_v \rightarrow \mathbb{R}$. Let x and y be two given points in images u and v , respectively. The structure tensors $T_u(x)$ and $T_v(y)$ define elliptical patches $B_{T_u}(x)$ and $B_{T_v}(y)$ around these points. Equations (9.3) and (9.4) suggest the following mapping between the elliptical patches:

$$P(x, y) = T_v(y)^{-\frac{1}{2}} R_v^{-1}(y) R_u(x) T_u(x)^{\frac{1}{2}}. \quad (9.8)$$

We can interpret $P(x, y)$ as a local affinity, mapping the elliptical patch associated with $T_u(x)$ into the one associated with $T_v(y)$. In other words, if u in the vicinity of x is an affinely transformed version of v in the vicinity of y , then $P(x, y)$ recovers the true affinity. Intuitively, an affine invariant patch similarity could be computed as the distance between the elliptical patch at y and the elliptical patch at x , transformed by $P(x, y)$ beforehand. In practice, it is more convenient to transform both elliptical patches to discs of radius r , as explained in Section 6.4, and to compare the aligned normalized patches:

$$\mathcal{D}^a(t, x, y) = \int_{\Delta_t} g_t(h) \left(u(x + T_u^{-\frac{1}{2}}(x) R_u^{-1}(x) h) - v(y + T_v^{-\frac{1}{2}}(y) R_v^{-1}(y) h) \right)^2 dh, \quad (9.9)$$

where Δ_t denotes a disc centered at the origin of \mathbb{R}^2 with radius proportional to scale $t > 0$ and big enough such that the weighting function g_t has effective support in Δ_t . If g_t is the Gaussian weighting function of variance t , the patch distance $\mathcal{D}^a(t, x, y)$ coincides with the similarity measure (4.17) derived in Section 4.4. If we replace the Gaussian weights by the bilateral weights as in Tomasi and Manduchi (1998), we recover the similarity measure (4.16). Recall that throughout this work we use the terms *patch similarity measure* and *patch distance measure* interchangeably.

While the parameter t in (9.9) affects the scale of patch comparison explicitly, the r parameter in (9.6) and (9.7) also has its impact. As discussed in

Chapter 7, we can link both parameters together by

$$t = \left(\frac{r}{\hat{t}} \right)^2,$$

where \hat{t} now defines the relative scale with respect to r . We call both t and \hat{t} scale parameters. In practice we specify \hat{t} because then it allows us to vary r independently.

In our notation $B_u(x)$ denotes a set of points constituting an affine covariant neighborhood. Even though we refer to it as patch, it does not represent actual color values. To explicitly specify a patch as a subset of color values of an image u we denote it as

$$p_u(x) := p_u(x, \cdot), \quad (9.10)$$

where

$$p_u(x, h) := u(x + T_u(x)^{-\frac{1}{2}}h)$$

and h belongs to Δ_t .

Let us finally note, that for color images we consider a generalization of (9.9). Let $\mathbf{u} : \Omega_u \rightarrow \mathbb{R}^3$ and $\mathbf{v} : \Omega_v \rightarrow \mathbb{R}^3$ be color images, then the affine invariant similarity measure becomes

$$\mathcal{D}^a(t, x, y) = \int_{\Delta_t} g_t(h) \left\| \mathbf{u}(x + T_u^{-\frac{1}{2}}(x)R_u^{-1}(x)h) - \mathbf{v}(y + T_v^{-\frac{1}{2}}(y)R_v^{-1}(y)h) \right\|_2^2 dh, \quad (9.11)$$

where $\|\cdot\|_2$ denotes the Euclidean norm of vectors in \mathbb{R}^3 . The affine covariant structure tensors and neighborhoods are computed using the corresponding gray-scale versions of \mathbf{u} and \mathbf{v} . From now on when referring to (9.9), we will implicitly assume (9.11) for color images.

In the following chapters we describe two applications of the similarity measure (9.9). For these specific applications it is sufficient and more practical to use the simpler similarity measure with Gaussian weights. The second similarity measure, involving approximated geodesic weights, might be useful in other applications.

Image Inpainting

In this chapter we present an application of the proposed affine invariant patch similarity measure (9.9) to exemplar-based image inpainting. Recall that for color images we implicitly consider (9.11) without changing the notation. In Section 10.1 we introduce the inpainting problem and in Section 10.2 review the related work. Section 10.3 is devoted to the variational formulation of the inpainting method that we propose. In Section 10.4 we discuss in detail the numerical implementation of the algorithm. Then in Section 10.5 we present some experiments asserting the validity of our theoretical approach together with a comparison with other well-known exemplar-based methods. Finally, Section 10.6 concludes the chapter.

10.1 Problem Statement

Image inpainting, also known as image completion or disocclusion, refers to the recovery of occluded, missing or corrupted parts of an image in a given region so that the reconstructed image looks natural. It has become a key tool for digital photography and movie post-production where it is used, for example, to eliminate unwanted objects that may be unavoidable during filming.

Automatic image inpainting is a challenging task that has received significant attention in recent years from the image processing, computer vision, and graphics communities. Remarkable progress has been achieved with the advent of exemplar-based methods, which exploit the self-similarity of natural images by assuming that the missing information can be found elsewhere outside the inpainting domain. Roughly speaking, these meth-

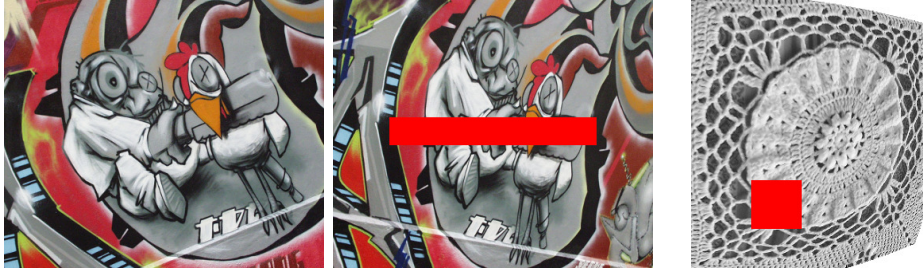


Figure 10.1: Self-similarity under different distortions. On the left: two views of the same scene related by a projective transformation. On the right: self-similar texture underwent a severe fish-eye lens distortion.

ods work by copying patches taken from the known part of the image and pasting them smartly in the inpainting domain. These methods can obtain impressive results but many of them rely on the assumption that the required information can be copied as it is, without any transformation. Therefore, applicability of such methods is limited to the scenes in which objects are in a fronto-parallel position with respect to the camera.

In the image formation process, textured objects may appear distorted by some complex transformation (see Figure 10.1). This is a pervasive phenomenon in our daily life. In fact, any person can mentally fill-in occluded parts of an image, even if the missing information is available to them under a different perspective. Our brain is able to appropriately transform the available information to match the perspective distortion of the occluded region. For instance, in Figure 10.1 one can easily infer what is hidden behind the red rectangle in the graffiti scene on the left, or use the non-trivially distorted context in the right image to fill-in the hole.

We address this issue by incorporating the affine invariant patch similarity measure (9.9) into a variational formulation of the inpainting problem. In this way we can compare patches, related by an affinity, and also transform known patches properly before pasting them in the inpainting domain. The transformation is determined for each patch in a fully automatic way. Moreover, instead of searching for an appropriate transformation in a high-dimensional space, our approach allows us to determine a single transformation from the surrounding texture content. As opposed to some previous works which only consider rotations and scalings, we can handle full affinities, which in principle extends the applicability of the method to

any transformation that can be locally approximated by an affinity, such as perspective distortion.

10.2 Related Work

Most inpainting methods found in the literature can be classified into two groups: geometry- and texture-oriented, depending on how they characterize the redundancy of the image.

The geometry-oriented methods formulate the inpainting problem as a boundary value problem and the images are modeled as functions with some degree of smoothness expressed, for instance, in terms of the curvature of the level lines (Masnou and Morel (1998); Ballester et al. (2001); Masnou (2002); Cao et al. (2011)), with propagation PDEs (Bertalmio et al. (2000)), or as the total variation of the image (Chan and Shen (2001)). These methods perform well in propagating smooth level lines or gradients, but fail in the presence of texture or big inpainting domains.

Texture-oriented (also called exemplar-based) methods were initiated by the work of Efros and Leung (1999) on texture synthesis. In that work the idea of self-similarity is exploited for direct and non-parametric sampling of the desired texture. The self-similarity prior is one of the most influential ideas underlying the recent progress in image processing and has been effectively used for different image processing and computer vision tasks, such as denoising and other inverse problems (Foi and Boracchi (2012); Buades et al. (2005); Gilboa and Osher (2008); Peyré (2009); Pizarro et al. (2010)). It has also found its application to inpainting: the value of each target pixel x in the inpainting domain can be sampled from the known part of the image or even from a vast database of images as in Hays and Efros (2007).

The exemplar-based approach to inpainting has been intensively studied (Demanet et al. (2003); Criminisi et al. (2004); Wexler et al. (2007); Kawai et al. (2009); Aujol et al. (2010); Arias et al. (2011)). However, many such methods are based on the assumption that the information necessary to complete the image is available elsewhere and can be copied *without any modification but a translation*.

Some works consider a broader family of transformations. In Drori et al. (2003) a heuristic criteria is used to vary the scale of patches. In Mansfield et al. (2011) and Barnes et al. (2010) the space of available patches is extended by testing possible rotations and scales of a source patch. The

search in the space of available patches is usually performed by a collaborative random search. However, this implies that for each query patch, the position of the matching patch as well as the parameters of the transformation (scale, rotation angle, tilt, etc) must be determined. The high dimensionality of the parameter space makes the search problem very computationally expensive and the excessive variability of candidates may lead to unstable results. In order to restrict the search space, the authors of [Cao et al. \(2011\)](#) propose to combine an exemplar-based approach that includes all rotated patches, with a geometric guide computed by minimizing Euler's elastica of contrasted level lines in the inpainted region.

Several authors (e.g., [Pavić et al. \(2006\)](#); [Huang et al. \(2013\)](#)) have addressed this issue using some user interaction to guide the search process. For instance, the user provides information about the symmetries in the image, or specifies 3D planes which are then used for rectification and the rectified planes in turn are used to look for correspondences. Recently, [Huang et al. \(2014\)](#) proposed a method for automatic guidance that searches for appropriately transformed source patches. It starts by detecting planes and estimating their projection parameters, which are then used to transform the patches. This allows one to handle perspective transformations, in situations when representative planes can be detected.

Most of those works use a similarity measure, either explicitly or implicitly, to compute a matching cost between patches. Our proposed affine invariant similarity measure (9.9) automatically distorts the patches being compared. Our method thus considers a rich patch space that includes all affine-transformed patches, furthermore, for each pair of patches the transformations are uniquely determined using the image content. This effectively limits the search space, making the method more stable. Since the patch distortions depend on the texture content of the image, our technique is related in that sense to a shape-from-texture approach ([Gårding \(1992\)](#); [Garding and Lindeberg \(1994\)](#); [Ballester and Gonzalez \(1998\)](#)).

In this chapter we extend the variational framework described, for example, in [Wexler et al. \(2007\)](#); [Kawai et al. \(2009\)](#); [Arias et al. \(2011\)](#). We propose a new energy and an optimization algorithm for affine invariant exemplar-based inpainting.

Let us finally note that [Wang \(2008\)](#) proposed a self-similarity measure for image inpainting, comparing dense SIFT descriptors on square patches of a fixed size. However, the method is not fully affine invariant, in particular, neither the dense SIFT descriptors nor the square patches are scale

invariant. Several authors have addressed the affine distortion and affine invariance problem in other contexts such as image comparison (Mikolajczyk and Schmid (2004)), object recognition (Matas et al. (2004)), and stereo (Garding and Lindeberg (1994)).

10.3 Inpainting Formulation

Exemplar-based inpainting methods aim at filling-in the image in such a way that each patch in the inpainting domain is similar to some known patch. This implies comparison of known patches with partially or completely unknown patches. For this we extend the variational framework described in Wexler et al. (2007); Kawai et al. (2009); Arias et al. (2011) by using the affine invariant similarity measure (9.9). We formulate the problem of inpainting from affinely transformed patches via the minimization of the following energy functional

$$E(u, \varphi) = \int_{\tilde{O}} \mathcal{D}^a(t, x, \varphi(x)) \, dx, \quad (10.1)$$

where $O \subset \Omega \subset \mathbb{R}^2$ is the inpainting domain, $\hat{u} : \Omega \setminus O \rightarrow \mathbb{R}$ is the known part of the image, \tilde{O} includes all the centers of patches intersecting O and \tilde{O}^c is its complement, that is, \tilde{O}^c contains centers of fully known patches (see Figure 10.2). The minimization of (10.1) aims at finding a visually plausible completion u of \hat{u} in the unknown region O . The additional variable $\varphi : \tilde{O} \rightarrow \tilde{O}^c$ determines for each unknown target patch the location of a source patch from which the information will be copied.

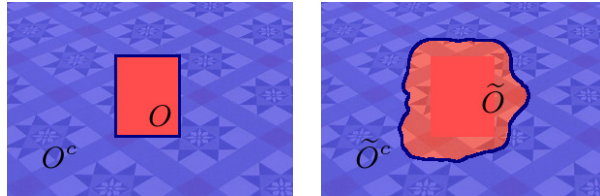


Figure 10.2: Schematic representation of the sets O , O^c , \tilde{O} and \tilde{O}^c .

This energy compares patches defined on elliptical regions centered at x and $\varphi(x)$. In the known part of the image, these regions are defined by the affine covariant structure tensors $T_{\hat{u}}$. Since the image is unknown inside the inpainting domain we have to estimate the structure tensors together with

the image. The relationship between u and T_u introduces a complex dependency in the energy (10.1), which complicates its minimization. Therefore, we propose to relax it and consider instead the minimization of the energy

$$\begin{aligned} \tilde{E}(u, \varphi, G) = & \int_{\tilde{O}} \int_{\Delta_t} g_t(h) \\ & \cdot \left(u(x + G(x)^{-\frac{1}{2}}h) - \hat{u}(\varphi(x) + T_{\hat{u}}(\varphi(x))^{-\frac{1}{2}}R_{\hat{u}}^{-1}(\varphi(x))h) \right)^2 dh dx \end{aligned} \quad (10.2)$$

where $G(x)$ is an invertible 2×2 matrix, $\forall x \in \tilde{O}$. For now, we will not restrict the tensor field G to be given by the structure tensors T_u . Instead, we consider it as an additional variable, in principle independent of u . In this way, we do not have to deal with the complex dependency between T_u and u . In practice, due to the properties of the affine covariant structure tensors, it turns out that the $G(x)$ can be estimated from $T_u(x)$ and the additional rotation $R_u(x)$, as will be explained later in this section.

Approximate Minimization Algorithm

We compute a local minimum of the energy with an alternating optimization scheme on the variables u , G and φ which is summarized in Algorithm 6.

Image update step. In the image update step, φ and G are fixed, and the energy is minimized with respect to u . With the change of variables $z = x + G(x)^{-\frac{1}{2}}h$, the Euler-Lagrange equation leads to the following expression:

$$\begin{aligned} u(z) = & \frac{1}{\varrho(z)} \int_{\tilde{O}} g_t \left(G(x)^{\frac{1}{2}}(z - x) \right) \\ & \cdot \hat{u} \left(\varphi(x) + T_{\hat{u}}(\varphi(x))^{-\frac{1}{2}}R_{\hat{u}}^{-1}(\varphi(x))G(x)^{\frac{1}{2}}(z - x) \right) \left| G(x)^{\frac{1}{2}} \right| dx, \end{aligned} \quad (10.3)$$

where $\varrho(z)$ is normalization factor such that the sum is an average. The field G determines elliptical patches centered at each $x \in \tilde{O}$. For each one of these patches a matching patch centered at $\varphi(x)$ is known, as well as its shape which is given by the structure tensor $T_{\hat{u}}(\varphi(x))$. The corresponding patch is then warped via the affinity

$$\tilde{P}(\varphi(x), x) = G(x)^{-\frac{1}{2}}R_{\hat{u}}(\varphi(x))T_{\hat{u}}(\varphi(x))^{\frac{1}{2}}$$

and aggregated in the inpainting domain. Note that if $G(x)$ is given by $T_u(x)$, then $\tilde{P}(\varphi(x), x)$ coincides with equation (9.8).

Affine correspondence update step. Given a fixed u , the minimization of the energy with respect to (φ, G) can be performed as independent minimization of the patch distance function \mathcal{D}^a for each $x \in \tilde{O}$. This problem is very complex to solve since it is a nearest neighbor search where we also optimize for the affine transformation of the patch at x , given by G .

We will exploit the properties of the affine covariant structure tensors to obtain an approximate solution. For that, let us consider a completion candidate u and assume that a local vicinity of x on u is an affinely transformed version of a local vicinity of $\varphi(x)$ on \hat{u} . That is, $u(x+h) = \hat{u}(\varphi(x) + Ah)$, which is the case when x and $\varphi(x)$ do actually correspond. Setting $G(x)$ such that $G^{-\frac{1}{2}}(x)R_{\hat{u}}(\varphi(x))T_{\hat{u}}(\varphi(x))^{\frac{1}{2}} = A$ will lead to a correct mapping and thus to the zero patch distance. On the other hand, using (9.8) we can find this affinity as $A = T_u(x)^{-\frac{1}{2}}RT_{\hat{u}}(\varphi(x))^{\frac{1}{2}}$ where R is some orthogonal 2×2 matrix and T_u is calculated on u . Then $G(x)$ such that $G^{\frac{1}{2}}(x) = R(x)T_u^{\frac{1}{2}}(x)$, together with $\varphi(x)$, will be global minimizers of the patch distance function \mathcal{D}^a at x . Therefore, we need to search only for $\varphi(x)$ and $R(x)$. An approximate $\varphi(x)$ can be found efficiently using our modified version of the PatchMatch algorithm of Barnes et al. (2009), detailed in Section 10.4. The additional rotation $R(x)$ is determined as described in Section 6.3, in the same way as for the known part of an image. Notice that for notation consistency we should write $R(x) := R_u(x)$.

Of course, if the neighborhood of x does not match any affinely transformed patch, then the estimated G might not minimize the patch distance \mathcal{D}^a .

Algorithm 6: Approximate minimization of $\tilde{E}(u, \varphi, G)$.

Input: Initial condition u^0 at O , tolerance $\tau > 0$

Output: Image completion u

repeat

 Compute affine covariant structure tensors $T_{u^{k-1}}(x)$ and rotations $R_{u^{k-1}}(x)$ for all $x \in \tilde{O}$;

 Estimate optimal correspondences φ^k using the modified PatchMatch (see Section 10.4);

 Update image $u^k = \arg \min_u \tilde{E}(u, \varphi^k, G^k)$, subject to $u^k = \hat{u}$ in O^c ;

until $\|u^k - u^{k-1}\| < \tau$;

Another interpretation of the approximate minimization can be given by

adding to the minimization of $\tilde{E}(u, \varphi, G)$ the constraint that $G^{\frac{1}{2}}(x) = R_u(x)T_u^{\frac{1}{2}}(x)$ for all $x \in \tilde{O}$ and for some rotation matrix $R_u(x)$, namely,

$$\min \tilde{E}(u, \varphi, G) \quad \text{subject to } G^{\frac{1}{2}} = R_u T_u^{\frac{1}{2}}.$$

The correspondence update step corresponds to the constrained minimization of the energy with respect to φ, G for a fixed image u . In the image update step the energy is minimized with respect to u , but without enforcing the constraint. Therefore, our approximate minimization can be seen as an alternating minimization applied to a constrained problem. The constraint is enforced only when minimizing with one of the variables (the pair φ, G). There are no theoretical guarantees for the convergence of such a scheme, although we have not yet encountered a practical case where the algorithm failed to converge.

10.4 Numerical Implementation

Image Update Step

The actual implementation of (10.3), that we use in our method, is

$$u(z) = \frac{1}{C(z)} \sum_{x \in \tilde{O}} g_t(T_u^{\frac{1}{2}}(x)(z - x)) m_c(x) w(x, z) \cdot \hat{u}(\varphi(x) + P(x, \varphi(x))(z - x)) |T_u^{\frac{1}{2}}(x)|, \quad (10.4)$$

where $P(x, \varphi(x)) = T_u^{-\frac{1}{2}}(\varphi(x))R_u^{-1}(\varphi(x))R_u(x)T_u^{\frac{1}{2}}(x)$ is the estimated affinity that maps the target patch at x onto the source patch at $\varphi(x)$. The structure tensor field T_u is computed using the inpainted image u from the previous iteration.

Of course, in the discrete setting some kind of interpolation needs to be done after transforming one elliptical patch into another by $P(x, \varphi(x))$. For that we use the Nadaraya-Watson estimator (Nadaraya (1964); Watson (1964)) with Gaussian kernel which standard deviation is specified by a free parameter σ_{NW} .

The extra term m_c in (10.4) is a so-called *confidence mask* that takes values from 1 to 0, exponentially decreasing with the distance to the set of known pixels O^c . This mask is usual in exemplar-based inpainting, for instance, it is used in Criminisi et al. (2004); Arias et al. (2011). It helps to guide the

flow of information from the boundary towards the interior of the inpainting domain, eliminating some local minima and reducing the effect of the initial condition. More precisely, we compute the confidence mask as

$$m_c(x) = (1 - c_0) \exp\left(-\frac{d(x, O^c)}{c_t}\right) + c_0, \quad (10.5)$$

where $d(x, O^c)$ is the distance from a point x to the boundary of the O^c set, such that $d(x, O^c) = 0$ when $x \in O^c$. Parameter $0 < c_0 \leq 1$ defines the smallest (asymptotic) value that m_c can take and $c_t > 0$ controls the rate of decay. In fact, c_t can be computed in such a way that the asymptotic value will always correspond to the maximum distance

$$c_t = -\frac{\max(d(\cdot, O^c))}{\ln c_0},$$

where $0 < c_0 < 1$. When $c_0 = 1$, the confidence mask is simply set to 1 everywhere and makes no effect. This confidence mask never changes during the inpainting process and can be precomputed for a given inpainting domain.

There is also another additional weighting term $w(x, z)$ in (10.4). In principle, all patches containing a pixel z contribute to its color value. To control the amount of contributors, we introduce the auxiliary Gaussian weight, that depends on the patch distance between a contributing patch and its corresponding known patch

$$w(x, z) = \exp\left(-\frac{(\mathcal{D}^a(t, x, \varphi(x)) - \min(\mathbf{D}(z)))^2}{2\sigma_{cut-off}^2(z)}\right), \quad (10.6)$$

where $\mathbf{D}(z) = \{\mathcal{D}^a(t, y, \varphi(y)) : z \in B_{T_u}(y)\}$ is a set of patch distances to known patches, computed among all patches contributing to z , and $\sigma_{cut-off}(z)$ defines a soft threshold for the patch distance values. This weight allows us to cut off contributors with low similarity (high distance) values, which in turn results in sharper reconstructions.

To compute $\sigma_{cut-off}(z)$ we begin by computing the first estimate for the cut-off distance

$$\mathcal{D}_1^a(z) = \gamma_{cut-off}(\max(\mathbf{D}(z)) - \min(\mathbf{D}(z))),$$

where $\gamma_{cut-off} \in (0, 1)$ is a parameter of the method. Since distance values are usually distributed unevenly, the initial distance threshold \mathcal{D}_1^a might discard

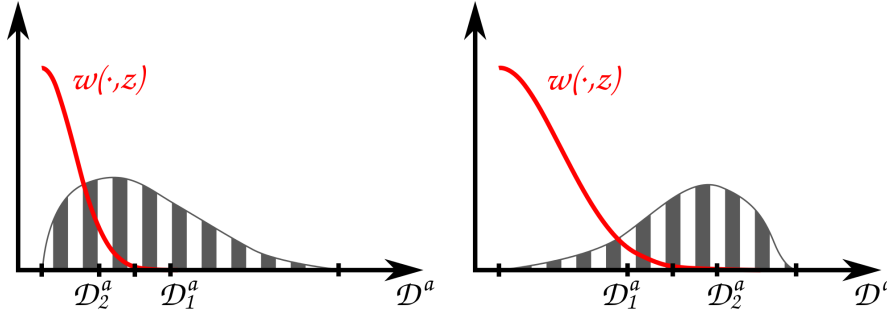


Figure 10.3: Schematic depiction of two different cases of patch distance distribution. Gray filled-in curves represent histograms of distance values computed between patches overlapping $z \in O$ and their most similar known counterparts from \tilde{O}^c . Weighting functions are shown in red. In both cases $\gamma_{cut-off} \approx 0.45$.

too few or too many contributors. Therefore, we very roughly estimate the density of values that fall below \mathcal{D}_1^a and refine the initial cut-off distance by

$$\mathcal{D}_2^a(z) = \gamma_{cut-off} \frac{\mathcal{D}_1^a |\mathbf{D}(z)|}{N_1},$$

where $|\mathbf{D}(z)|$ stands for the total number of elements in the set $\mathbf{D}(z)$ and $N_1 = |\{\mathcal{D}^a \in \mathbf{D}(z) : \mathcal{D}^a - \min(\mathbf{D}(z)) < \mathcal{D}_1^a(z)\}|$ is the number of distance values retained by \mathcal{D}_1^a . Then the final cut-off threshold is given by

$$\sigma_{cut-off}(z) = \frac{1}{6} (\mathcal{D}_1^a(z) + \mathcal{D}_2^a(z)).$$

The factor $\frac{1}{6}$ in the formula above implies that the Gaussian (10.6) approaches zero at the average cut-off distance between the first and second estimates. Figure 10.3 illustrates the cut-off distances for two different cases of distance values distribution.

The energy (10.1) is non-convex and might have several local minima. As a consequence, there is a dependency on the initialization. To alleviate this dependency, we try to promote the propagation of information from the boundary towards the interior of the inpainting domain during the very first iterations of inpainting. Recall that the extended domain \tilde{O} contains the centers of all elliptical patches overlapping the inpainting domain. We enlarge \tilde{O} by a few pixels to capture a narrow stripe $\tilde{O}^+ := (\tilde{O} \oplus B_b) \setminus \tilde{O}$ around the extended inpainting domain, that contains centers of completely known elliptical patches. Here B_b denotes a circular structuring element

of radius b , acting in the dilation of \tilde{O} . Obviously, the elliptical patches in \tilde{O}^+ do not intersect the inpainting domain. To make them contribute to the inpainting we should enlarge them first. For that we recompute them doubling the value of r . Notice that we use $r' = 2r$ only for the points within the stripe \tilde{O}^+ and only in the image update step. We do not recompute the corresponding structure tensors; thus, we only increase the sizes of these elliptical patches and do not modify their shapes. This additional contribution from elliptical patches, that do not depend by any means on the inpainting domain, boosts the information propagation at the boundaries of the inpainting domain.

Affine Correspondence Update Step

During the update of the correspondence map we compute an approximation of the nearest neighbor field using PatchMatch of Barnes et al. (2009,

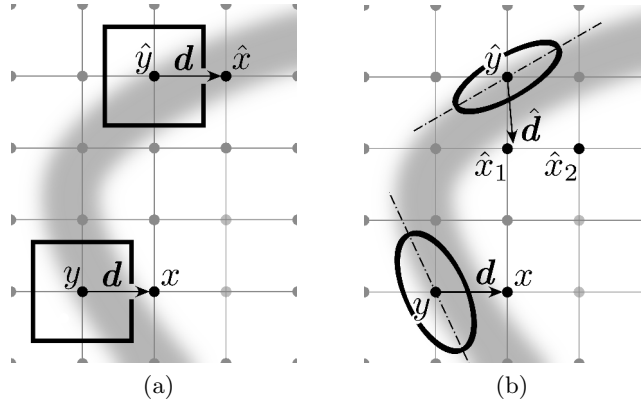


Figure 10.4: Propagation directions in PatchMatch: (a) original scheme, (b) modified scheme.

2010). The PatchMatch algorithm speeds up the computation of optimal correspondences by exploiting the correlation between patches so that they can be found collectively. Since we are working with elliptical patches which might be arbitrarily rotated, we adapt the PatchMatch propagation scheme to take this into account. Let x be the current pixel and $d_1 = (\pm 1, 0)$, $d_2 = (0, \pm 1)$ be the directions of propagation. Then, the adjacent pixels $y_i = x - d_i$ ($i = 1, 2$) are tested during the propagation. Assume $i = 1$ (see Figure 10.4). Pixel $\hat{y} = \varphi(y)$ is the current nearest neighbor candidate for y . The standard PatchMatch would try to propagate position $\hat{y} + d$ to pixel x

(Figure 10.4a). In contrast, we calculate the direction $\hat{d} = P(y, \hat{y})d$, where P is the a priori connection, and we try a few positions along that direction (Figure 10.4b). This generalization gives more meaningful propagation along edges.

At early iterations of our algorithm, the inpainted image may be blurry. This is typical for iterative patch-based inpainting methods and is caused by aggregating patches that do not coincide in their overlap area at the initial iterations. As discussed in Section 5.3, the structure tensors are sensitive to blur, tending to larger elliptical patches in blurry regions. Essentially, smoothing of an image suppresses small details and produces the same effect as scaling the image down. The elliptical patches in turn capture larger areas. To compensate for this, we allow the parameter r to vary during the correspondence map estimation. That is, while $T_{\hat{u}}(\varphi(x))$ is always computed with the fixed r , say r_0 (a given parameter of the method), in the computation of $T_u(x)$ we consider a few values of r smaller than r_0 and select the one giving the smallest patch distance \mathcal{D}^a between points x and $\varphi(x)$. The value of r decreases by the step of h_r and the number of such steps is specified by n_r parameter. Let us note, that to be able to compare patches, computed with different values of r , we scale the normalized patches to discs of radius one.

10.5 Experimental Results

In this section we present results obtained by the proposed method. First of all we summarize the parameters of the method and specify the values used in our experiments:

- r – the “radius” used in the elliptical patch calculation (9.2). See Sections 5.1 and 9.1. Is set to 250 – 350 in our experiments.
- \hat{t} – the “scale” that controls the intra-patch Gaussian weights. See Section 9.1. Is set to 3.0 – 3.5 in our experiments.
- g – the resolution of the regular grid used for interpolation during the patch distance calculation. See Section 6.4. Is set to 21 in our experiments.
- τ – the tolerance in the energy minimization process. See Section 10.3. Is set to 0.01 in our experiments.

- b – the width of the stripe \tilde{O}^+ in pixels. See Section 10.4. Is set to 6 in our experiments.
- $\gamma_{cut-off}$ – the soft threshold on the percentage of contributing patches. See Section 10.4. Is set to 0.4 in our experiments.
- c_0 – the asymptotic value of the confidence mask. See Section 10.4. Is set to 0.1 in our experiments.
- σ_{NW} – the interpolation coefficient used in the image update step. See Section 10.4. Is set to 0.3 in our experiments.
- n_{PM} – the number of PatchMatch iterations. See Barnes et al. (2009). Is set to 16 in our experiments.
- h_r, n_r – the length of the steps in r and the number of such steps, used in the $T_u(x)$ computation to compensate blurring. In our experiments h_r is set to 10 – 15 and n_r is set to 4 – 9. See Section 10.4.

Another implicit parameter, that can be found in many exemplar-based inpainting methods, is the initialization of the inpainting domain. In the very first iteration we do not expect to find any good matches for the patches from the middle of the inpainting domain and rely on the mostly known patches. Since in the absence of texture elliptical patches tend to grow in area, we do not use the common initialization by a uniform color. Instead we initialize the inpainting domain using a smooth checkerboard pattern to keep patches inside it small.

For all the experiments in this section, we compare our results with the ones obtained by the multiscale Non-Local Means method (Wexler et al. (2007); Kawai et al. (2009)) which we find to be a representative exemplar-based image inpainting method operating with only translations of patches. Whenever possible, we also compare against the method of Mansfield et al. (2011) with a single scale and considering rotations, and the method of Huang et al. (2014). In both cases we use implementations provided by the authors.

As a sanity check we first test the proposed method on a synthetic example, displayed in Figure 10.5. We take a textured image and create an affinely transformed version of it. We select a part of the transformed image as the inpainting domain. Instead of using the rest of the transformed image to copy information from, we make *the original (not transformed) image to be the source*. Let us remark that the ground truth affinity is not provided



Figure 10.5: First row: source image, target image with the inpainting domain shown in red, and close-ups around the inpainting area of the NL-Means result and the result of our method. Second row: evolution of the inpainting domain over iterations of our method (every second iteration).

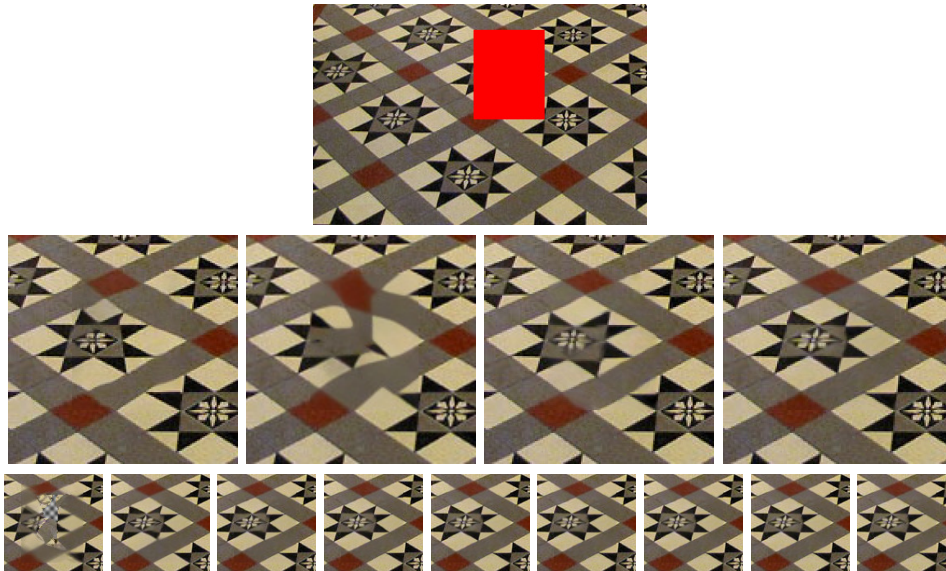


Figure 10.6: First row: image with the inpainting domain shown in red. Second row: close-ups around the inpainting area of the NL-Means result, the result of Mansfield et al. (2011) (considering rotations), the result of Huang et al. (2014), and the result of our method. Third row: evolution of the inpainting domain over iterations of our method (every third iteration).

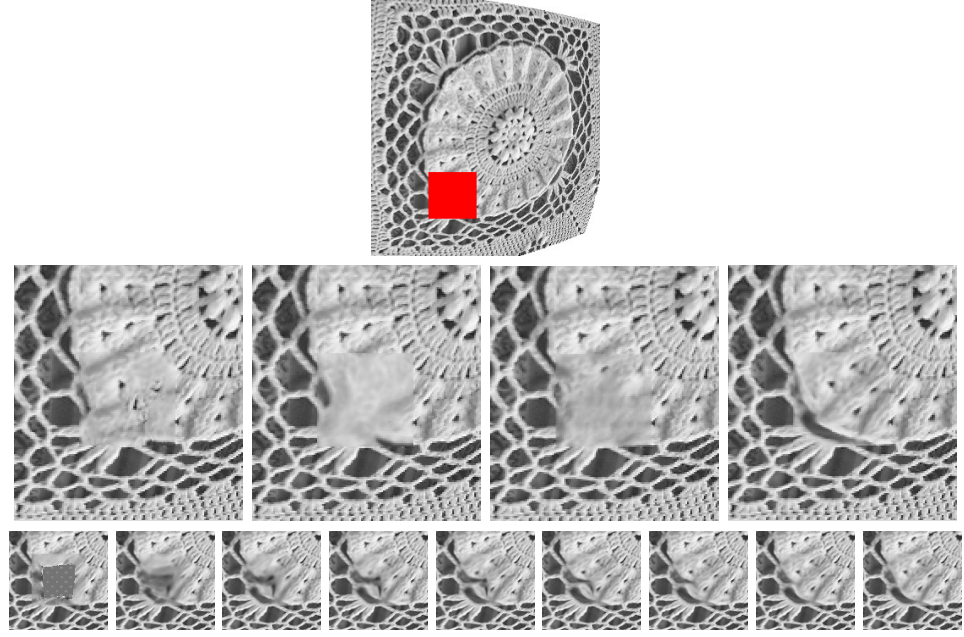


Figure 10.7: First row: image with the inpainting domain shown in red. Second row: close-ups around the inpainting area of the NL-Means result, the result of Mansfield et al. (2011) (considering rotations), the result of Huang et al. (2014), and the result of our method. Third row: evolution of the inpainting domain over iterations of our method (every third iteration).

to the algorithm, hence, we test the ability of the proposed method to identify and copy affinely transformed patches. We do not show any results for Mansfield et al. (2011) and Huang et al. (2014) for this experiment, since the available implementations do not support the use of a separate image as a source.

A more realistic case would be associated with a more general transformation. Since for planar objects a projective transformation can be locally approximated by an affinity, in the second example (shown in Figure 10.6) we test the robustness of our method in the reconstruction of an image distorted by perspective. As usual in inpainting applications, in this experiment we use *the known part of the image as source*. We compare our method with the Non-Local Means method, that works only with translations, and additionally with the method of Mansfield et al. (2011) in the mode when the rotations are also considered, and the method of Huang



Figure 10.8: First row: source image, target image with the inpainting domain shown in red, and close-ups around the inpainting area of the NL-Means result and the result of our method. Second row: evolution of the inpainting domain over iterations of our method (every second iteration).

et al. (2014). Note that the latter method successfully determines a single plane in the image and, as expected, achieves a good reconstruction.

The third example (Figure 10.7) demonstrates the reconstruction of a texture with some lens distortion applied to it. *The known part of the image is used as a source* and, like in all other experiments, just a rotation of source patches is not sufficient to obtain a good result. As in the previous case, here we compare our method with the Non-Local Means method (translations), the method of Mansfield et al. (2011) (translations and rotations), and the method of Huang et al. (2014) (projective transformation).

A final experiment, which is also potentially interesting for real applications, consists in inpainting one view of a scene using information from another view of the same scene. Figure 10.8 shows the results of this experiment where we have applied the proposed method to two views related by an unknown homography. As before, we compare our result with the result of the Non-Local Means method.

Let us note that the method of Mansfield et al. (2011) also supports rotations plus scalings. However, we could not obtain meaningful results on these examples for this mode. It seems that the additional variability added by the scalings makes it easier for the algorithm to be trapped in a bad local minimum. For example, a constant region can be produced by scaling a small uniform patch.

Finally, we briefly discuss the limitations of the proposed method. Since the transformation between two patches is estimated from the surrounding

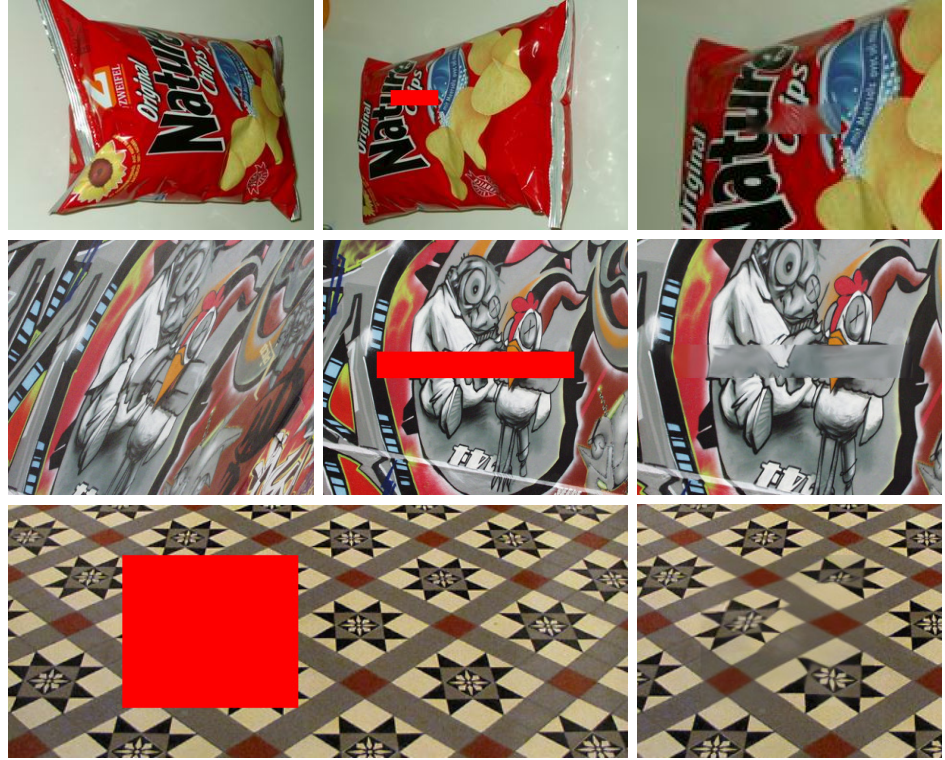


Figure 10.9: Failure cases. From top to bottom: insufficient textural information, severe distortion, too big inpainting domain. More details are given in the text.

texture, the method fails when there is not enough textural information (Figure 10.9, first row). Severe transformations between pairs of patches may be recovered incorrectly. This can be illustrated by replacing the source image in the last experiment with a much more slanted view (Figure 10.9, second row). The proposed method does not exploit the common multiscale scheme which limits the maximum possible size of the inpainting domain (Figure 10.9, third row).

10.6 Conclusions

In this chapter we have presented a new variational formulation for exemplar-based image inpainting that takes advantage of affine invariant self-similarities of real images. Due to the affine invariant similarity measure (9.9),

incorporated in the inpainting energy functional, our method naturally operates with the shape-adaptive patches. Since an appropriate affine transformation relating any given pair of patches is computed locally, the method can handle textured objects under projective transformations or even more complex distortions. For instance, it is well suited for inpainting of one view of a scene using other view of the same scene as a source, or for completion of an image acquired with a so-called fisheye lens. The obvious requirement though is to have enough textural information.

The exemplar-based inpainting is a complex non-convex problem with many local minima. As pointed out in [Cao et al. \(2011\)](#), additional degrees of freedom in transformations of patches make it even more complex. The proposed similarity measure is beneficial in this respect, because shapes of patches are uniquely determined by an underlying texture and the amount of ways in which one patch can be aligned with another is constrained. This eliminates some of the variability and leads to a faster and more accurate minimization algorithm. We have proposed a tractable approximate optimization scheme for our formulation of the inpainting problem and then have delved into the details of its numerical implementation. Finally, we have shown inpainting results for several different settings together with some failure cases.

Some weaknesses of the method are intrinsic to the proposed approach itself. For example, the lack of textural information makes shape-adaptation and alignment of patches unreliable. As concerning large inpainting domains, the issue can be solved by exploiting a multiscale scheme, as it is customary in the literature ([Wexler et al. \(2007\)](#); [Kawai et al. \(2009\)](#); [Arias et al. \(2011\)](#)). Modifying the proposed method to be multiscale, however, is a complex problem in itself and it is left outside the scope of the current work as an interesting direction for future research.

Image Denoising

In this chapter we demonstrate how the well-known Non-Local Means image denoising method can be extended with the help of the affine invariant patch similarity measure (9.9) or its generalization for color images (9.11). Our extension takes advantage of the affine invariant self-similarities in real images and thus is able to aggregate more similar patches which leads to better denoising results. At first we introduce the denoising problem in Section 11.1 and in Section 11.2 we review the related work on image denoising. Then in Section 11.3 we explain the proposed extension and in Section 11.4 we discuss the most important implementation details. In Section 11.5 we briefly comment on the choice of parameters of the method. In Section 11.6 we present both quantitative and qualitative assessments of the proposed method. Finally, Section 11.7 concludes the chapter.

11.1 Problem Statement

There are many different ways of obtaining a digital image of a real world scene: digital cameras, scanning devices, etc. Unless an image is generated completely by a computer, it is subject to the acquisition noise. Many different kinds of denoising algorithms are widely used to suppress this noise while preserving image details, textures and colors. They are embedded in firmware of digital cameras and are also applied in the very beginning of image and video processing pipelines.

In this chapter we focus on the Non-Local Means framework for image denoising and propose an extension for it, which effectively exploits the affine invariant self-similarities present in images of real scenes. The Non-Local

Means approach to image denoising considers 2D patches and, in general, assumes that for every noisy reference patch a bunch of other similar noisy patches can be collected together and averaged to produce a single denoised patch. Usually, the contribution of each patch in the averaging is additionally weighted by its similarity to the reference patch. The bigger amount of samples is aggregated, the more noise can be suppressed by averaging. Therefore, the ability to find many similar patches largely determines the performance of the Non-Local Means denoising. The original method of Buades et al. (2005) operates with square patches of a fixed size and thus is able to collect similar patches that differ only in their location in an image. However, in real images similar patches might be related by much more complex transformations, for example, by an affinity. Such distinction in appearance of patches, that contain essentially the same visual information, might be due to some nontrivial self-similarity of a scene or simply because of difference in positions with respect to the camera. With the help of the affine invariant patch similarity measure (9.9) we can properly compare patches related by an affinity. As a result, more similar patches can be found and aggregated, which in turn provides better denoising results. We demonstrate that our method outperforms the original Non-Local Means method both in terms of PSNR values and in visual quality of results. Compared with a state-of-the-art method of Lebrun et al. (2013b), our method concedes in numbers, but produces images that visually look at least as good and in some cases even more pleasing.

11.2 Related Work

The problem of removing the noise of an image while preserving its structure and characteristics like geometry, texture or color has been extensively investigated over the last decades. Some early works approach denoising by local methods, for example, using anisotropic diffusion as in Perona and Malik (1990b). The total variation denoising model of Rudin et al. (1992) inspired many variational formulations for image denoising. In Rudin et al. (1992) it was applied for images corrupted by the Gaussian noise (see also Chambolle (2004); Almansa et al. (2006)), and has also been used for Riccian (Martn et al. (2011)), salt-and-pepper (Nikolova (2004)) and multiplicative noise (Aubert and Aujol (2008)). Other operators have been proposed, for example, the total generalized variation (Bredies et al. (2010)) or non-local regularizers as in Gilboa and Osher (2008); Brox et al. (2008). Frequency domain methods have been developed and extensively studied as well, for

instance, windowed discrete cosine transform (DCT) in Yaroslavsky (1996); Yu and Sapiro (2011), wavelet transform based methods in Donoho and Johnstone (1994); Portilla et al. (2003), curvelets in Starck et al. (2002), etc. More recently it was proposed (Elad and Aharon (2006); Mairal et al. (2008); Yu et al. (2010)) to learn dictionaries for sparse and redundant image representation from images themselves instead of exploiting predefined ones (like wavelets or DCT). These methods attempt to model the space of image patches.

Non-local or patch-based approaches are used in most of the state-of-the-art methods for image denoising, restoration, super-resolution, inpainting and object recognition (Peyré (2009); Protter et al. (2009); Pizarro et al. (2010); Foi and Boracchi (2012); Pierazzo et al. (2014); Kheradmand and Milanfar (2014)). Image denoising has gone along with the advances in patch-based techniques. Non-Local Means of Buades et al. (2005) exploits the notion of non-local self-similarity of image patches and works by collecting and averaging the most similar patches in the image. This seminal approach inspired many denoising methods and has also led to the study and development of non-local regularity priors (Peyré et al. (2008); Arias et al. (2011)). One of the state-of-the-art denoising methods, the BM3D method of Dabov et al. (2007), combines spatial self-similarity with thresholding in the frequency domain through the collaborative filtering of groups of similar patches. The resembling method of Zhang et al. (2010) exploits principal component analysis instead of DCT. Both methods involve two almost identical steps: the first one provides the basic estimate and the second one refines the final result using the initial estimate as a guide. A Bayesian interpretation of the method of Zhang et al. (2010), named Non-Local Bayes, was recently proposed in Lebrun et al. (2013b). Non-Local Bayes is a spatial-based method that improves Non-Local Means by considering a Gaussian probability model for each set of similar patches.

In many denoising works it is assumed that the noise type and level are known in advance. The Noise Clinic method of Lebrun et al. (2015) for blind denoising goes beyond this assumption and combines a noise model estimation with a multiscale modification of the Non-Local Bayes method.

Let us finally note that rotationally invariant patch comparison for image denoising was already studied in the past (Grewenig et al. (2011); Zuo et al. (2015)). Besides that, it was noticed that denoising with regular square patches may cause noise halos around contrasted edges. To remove this kind of artifacts it was proposed in Deledalle et al. (2012) to replace square

patches by several types of neighborhoods with various shapes (discs, bands, half-discs, quarter-discs) and to combine their estimations using Stein's Unbiased Risk Estimate (Stein (1981)). Our affine invariant patch similarity measure (9.9) considers a richer domain of shape-adaptive patches. On the other hand, for each pair of patches the transformations are determined using the image content, which effectively reduces the search space when looking for similar patches.

11.3 Affine Non-Local Means Denoising

In this section we describe our extension of the well-known Non-Local Means denoising method. We start by motivating the use of the proposed affine invariant similarity measure (9.9) for patch-based denoising. Additionally, we propose a modification that better adapts this similarity measure for denoising.

Motivation

Recent progress in image denoising is associated with the concept of self-similarity which is pervasive for natural images. In a broad sense, such methods as Buades et al. (2005); Lebrun et al. (2013b); Dabov et al. (2007); Kheradmand and Milanfar (2014) work by combining neighborhoods of points or patches of a noisy image that appear similar to each other and extracting from them a common estimate of the original noiseless image. The basic principle is that, by the variance law in probability theory, the noise standard deviation of the average of samples decreases. This approach is intuitively consistent, if one can find many samples for every image detail. In order to find patches which are most similar to a reference patch, many denoising methods use a simple patch distance measure:

$$\mathcal{D}(x, y) = \frac{1}{|B|} \sum_{i \in B} (u(x + i) - v(y + i))^2, \quad (11.1)$$

where B denotes a neighborhood of the origin $0 \in \mathbb{R}^2$ forming a patch, and $|B|$ denotes the area of B (in practice, the number of pixels in B). This can be rewritten in the continuous setting as:

$$\mathcal{D}(x, y) = \frac{1}{\varrho} \int_{\mathbb{R}^2} \eta(h) (u(x + h) - v(y + h))^2 dh, \quad (11.2)$$

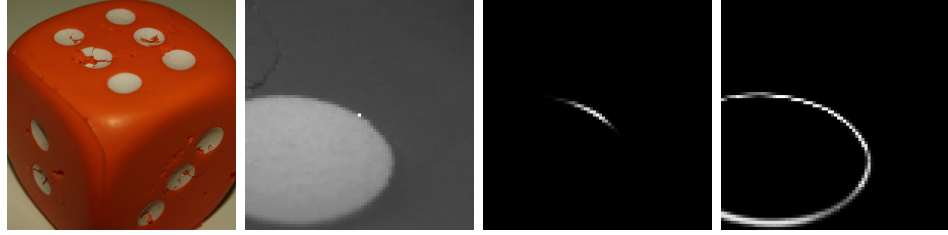


Figure 11.1: From left to right: original Dice image, a zoom of a search window with its central (reference) point shown in white, similarity values computed between the central point and all other points using the similarity measure (11.2) and square patches, and using the proposed affine invariant patch similarity measure (9.9). White color encodes the highest similarity.

where ϱ is the normalizing factor, $\eta(h)$ is a characteristic function of the patch which equals to 1 when $h \in B$ and 0 otherwise. Sometimes η is a weighted characteristic function such as a Gaussian.

Usually, patches are set to be squares and no transformations are allowed between two patches being compared. Since it is numerically impractical to check the whole image, similar patches are searched in a relatively small window around a reference point. The usual size of the search window ranges from 21 by 21 to 35 to 35 pixels. This allows one to reduce the overall computational expenses; however, when the level of noise increases, it is becoming less probable to encounter enough similar patches. If we allow for more complex transformations between patches being compared, we can increase the amount of similar patches within the search window and thus improve the denoising result. Figure 11.1 illustrates this argument with a simple experiment on the classical Dice image. For a given point and a search window around it we calculate the similarity between that point and all other points in the window using both the similarity measure (11.2) and our proposed similarity measure (9.9). As can be seen, in our case much more points along the edge receive high similarity values. This in turn leads to a better denoising result, as will be shown in Section 11.6.

Description of the Method

In this work we follow the strategy of the original Non-Local Means scheme proposed in Buades et al. (2005). In particular, given a noisy image we go through all of its points and, for every point, we calculate patch distances

between that reference point x and other points y around it. As in the other works, we limit this set of surrounding points to a square window of a given size w , centered at the reference point x . Instead of the very common patch distance measure (11.2) we use the proposed measure (9.9). Consequently, instead of common square patches, we consider elliptical patches given by the affine covariant structure tensor at every point of an image. Note that in the denoising problem, v in (9.9) coincides with u .

In order to convert distances into similarities we use the exponential function and define the similarity of the patches centered at x and y as:

$$\mathcal{S}^a(x, y) := \exp\left(-\frac{\mathcal{D}^a(t, x, y)}{\lambda^2}\right), \quad (11.3)$$

where $\lambda = b\sigma$ is a bandwidth that depends on the standard deviation σ of the noise and is controlled by the parameter $b > 0$ of the method. Notice that the distance \mathcal{D}^a is already squared.

To denoise an elliptical patch at the reference point x , we average all the patches at surrounding points y using their similarity values $\mathcal{S}^a(x, y)$ as weights. Note that in this aggregation process elliptical patches have to be appropriately transformed to match the patch at x being denoised. We calculate this transformation from the structure tensors using the a priori connection (9.8). The denoised patch is then given by a weighted average

$$\begin{aligned} \hat{p}_u(x) &= \frac{1}{\varrho(x)} \sum_{y \in W_w(x)} \mathcal{S}^a(x, y) (P(y, x) p_u(y)), \\ \varrho(x) &= \sum_{y \in W_w(x)} \mathcal{S}^a(x, y), \end{aligned} \quad (11.4)$$

where $W_w(x)$ denotes the square search window of size w , centered at x . Recall that $p_u(x)$ is defined as (9.10) and stands for a patch as a subset of color values of u . The notation $P(y, x)p_u(y)$ colloquially designates the mapping of an elliptical patch at y to an elliptical patch at x by $P(y, x)$. The map $P(y, x)$ first transforms an elliptical patch to a disc, then rotates this disc and finally transforms it into another elliptical patch. Of course, in the discrete setting, some kind of interpolation of color values has to be done after this transformation.

Since denoised patches may overlap each other, every pixel of the resulting image receives multiple color estimates. The estimates coming from a denoised patch are weighted depending on their distance to the center of that

patch. The resulting color for a pixel is therefore calculated as a weighted average of all estimates. We use here an anisotropic Gaussian kernel, given by the affine covariant structure tensor, as an intra-patch weighting function

$$g_t(T_u, x) = \exp \left(-\frac{\langle T_u(x)(y-x), (y-x) \rangle}{2t} \right),$$

where $y \in B_u(x)$ and $t = (r/\hat{t})^2$. Essentially, these are the same weights as in the similarity measure (9.9) and we use the same value of \hat{t} in both cases. In this way the patch aggregation complies with the patch comparison.

The proposed method described above is summarized in Algorithm 7.

Algorithm 7: The extension for Non-Local Means image denoising.

```

Input:  $u : \Omega_u \rightarrow \mathbb{R}^3$  // noisy image
Output:  $\hat{u} : \Omega_u \rightarrow \mathbb{R}^3$  // denoised image

Compute tensor field  $T_u$ 
 $\hat{u} \leftarrow 0$ 
 $\varrho_{\hat{u}} \leftarrow 0$ 
for  $x \in \Omega_u$  do
     $\hat{p}_u(x) \leftarrow 0$  // denoised elliptical patch
     $\varrho(x) \leftarrow 0$ 
    for  $y \in W_w(x)$  do
         $\hat{p}_u(x) \leftarrow \hat{p}_u(x) + \mathcal{S}^a(x, y)(P(y, x)p_u(y))$ 
         $\varrho(x) \leftarrow \varrho(x) + \mathcal{S}^a(x, y)$ 
     $\hat{p}_u(x) \leftarrow \varrho(x)^{-1} \hat{p}_u(x)$  // normalize  $\hat{p}_u(x)$ 
    Compute intra-patch Gaussian weights  $g_t(T_u, x)$ 
     $\hat{u} \leftarrow \hat{u} + g_t(T_u, x) \hat{p}_u(x)$ 
     $\varrho_{\hat{u}} \leftarrow \varrho_{\hat{u}} + g_t(T_u, x)$ 
 $\hat{u} \leftarrow \varrho_{\hat{u}}^{-1} \hat{u}$  // normalize  $\hat{u}$ 

```

Patch Size Constraint

In most of the patch-based methods square patches of a fixed size are used. The size of the patches may change depending on the noise level; however, it is always the same within a single image. Commonly square patches are chosen to be small, their size ranges from 3 by 3 to 7 by 7 pixels (sometimes slightly bigger). Our method allows the shapes of patches to adapt to the local image content; thus, the patches may well have different sizes. We



Figure 11.2: Elliptical patches in images with different levels of noise. From left to right, standard deviation of noise: 0 (original noiseless image), 2, 5, 10, 15.



Figure 11.3: Elliptical patches in the original noiseless image, calculated using different size constraints (different values of ρ_{max} in (11.8)). From left to right, ρ_{max} : ∞ ($\beta = 0$, no size constraint), 20, 15, 10 and 5.

would like to preserve this shape-adaptiveness and at the same time to be able to limit the maximum patch size to capture only small pieces of visual information. To some extent the r parameter in (9.2) controls the size of a covariant elliptical patch; however, it also depends on the image content itself (see Figure 11.2, left image). For the same value of r , elliptical patches are always significantly bigger in homogeneous regions than in textured regions or close to edges. As shown in Figure 11.2, in the presence of noise elliptical patches tend to shrink due to the additional gradients induced by the noise. This effect is accentuated in homogeneous regions; therefore, elliptical patches become more uniform in size. On the other hand, as these additional gradients have random magnitudes and directions, the *shape* of patches does not change a lot. This phenomenon motivates the way of placing an upper boundary constraint on the size of elliptical patches.

We are interested in simulating additive Gaussian noise in our structure tensor computation scheme (9.5)–(9.7). From this perspective, additional noise can intuitively be seen as a multitude of extra gradients with random directions and expected magnitude related to the variance of noise. We approximate it by adding a small constant value to the diagonal elements and define a new structure tensor to be

$$\tilde{T}_u(x) = T_u(x) + \beta I, \quad (11.5)$$

where β controls the amount of simulated noise and, therefore, the maximum allowed size of the associated elliptical patches, say it $\tilde{B}_u(x)$. Notice that the constraint term βI is incorporated right into the iterative scheme for structure tensor construction; thus, equation (9.5) becomes

$$T_u^{(k)}(x) = \frac{\int_{B_u^{(k-1)}(x,r)} Du(y) \otimes Du(y) dy}{\text{Area}(B_u^{(k-1)}(x,r))} + \beta I. \quad (11.6)$$

To relate β with the size constraint, let the reference point x be located in an infinitely wide region of uniform color. Since all the gradients are equal to zero, the term $T_u(x)$ is equal to zero as well and the resulting structure tensor is defined solely by β . If $\beta = 0$, an elliptical patch at x would have infinite size. If $\beta > 0$, the patch becomes a disc. From (9.2), its boundary is given by

$$\partial \tilde{B}_u(x) = \{y : \langle \beta I(y-x), (y-x) \rangle = r^2\} = \{y : \beta \|y-x\|^2 = r^2\}. \quad (11.7)$$

Let ρ_{max} be the radius of this disc. Then $\|y-x\|^2 = \rho_{max}^2$ and

$$\beta = \frac{r^2}{\rho_{max}^2}. \quad (11.8)$$

The relation (11.8) allows us to parametrize the patch size constraint in terms of the maximum possible radius of a patch, shall it appear in a completely homogeneous region. Figure 11.3 shows the effect of applying the size constraint with different values of ρ_{max} . Note that for $\rho_{max} = 5$, elliptical patches are almost uniform in size and still follow the image content.

Of course, this term, added to the structure tensors, breaks their affine covariance property. Most strongly it affects invariance of the similarity measure to scaling. This is easy to verify mathematically and can be illustrated with the following experimental example (Figure 11.4). Let two corresponding points be given in two images related by an affinity, in this particular case – a scaling. Two elliptical patches calculated using the original scheme are shown in Figures 11.4a and 11.4b. They cover the same visual information. Elliptical patches with the size constraint applied are shown in Figures 11.4c and 11.4d. The patch shown in Figure 11.4c cannot grow enough to take the edges into consideration. As a consequence, the corresponding elliptical patches do not capture the same visual information; moreover, tensors do not recover the correct local affinity. For some applications this breaking of affine covariance is unacceptable; however, in

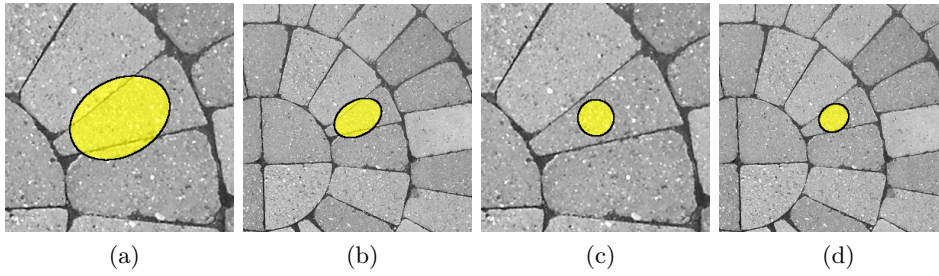


Figure 11.4: Effect of constraining the patch size. (a) and (c) show part of an image in its original size, (b) and (d) show part of the same image scaled down by 2. (a) and (b) illustrate elliptical patches without any size constraints, (c) and (d) illustrate elliptical patches with the size constraint $\rho_{max} = 25$ applied.

the case of denoising we do not expect to encounter severe change in scale within a relatively small search window.

The size constraint plays the most important role for the cases of small levels of noise when it prevents elliptical patches from growing too big in uniform regions. When there is a clearly defined structure within a patch it is less affected by the size constraint.

Homogeneous Region Test

Our patch comparison by means of the affine invariant similarity measure can be related to the shape-from-texture techniques in the sense that the shape of elliptical patches is defined by the underlying image texture. Thus, in the absence of clear textural information in homogeneous regions, the shape of elliptical patches becomes very dependant on the image noise. In the denoising application this leads to insufficient suppression of high frequency noise in such regions. To reduce this dependency, the parameters of the method can be adjusted to allow for bigger patches; however, it usually leads to a blurry reconstruction in textured regions. To avoid contradiction in the choice of parameters for denoising homogeneous regions and fine image details, such as textures and standalone edges, we treat them differently.

Motivated by the approach of Lebrun et al. (2013b), we first check whether a reference patch can be considered as located in a homogeneous region. For that we calculate the variance of all color values in a set of n_H patches in $W_w(x)$, most similar to the reference patch at x . Here n_H is a parameter

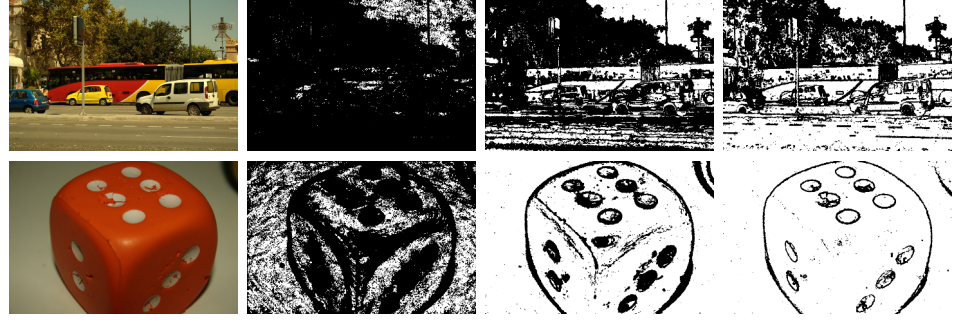


Figure 11.5: Left to right: original images, and homogeneous regions, in white, computed for noisy images with noise standard deviation 2, 10 and 30.

of the method. As commented in [Lebrun et al. \(2013b\)](#), for homogeneous regions that color variance is expected to be small and close to the noise variance. We check if it is less than $\gamma_H \sigma^2$, where σ is the noise variance and γ_H is a thresholding parameter of the method. Note that we use the same values for both n_H and γ_H in all our experiments (see Section 11.5). If the homogeneous region test is passed, meaning that the patch belongs to a homogeneous region, the reference patch is denoised by filling it with a single average color $c(x)$, computed from the similar patches. For every channel separately:

$$\begin{aligned}
 c(x) &= \frac{1}{\varrho(x)} \sum_{y \in \mathcal{P}(x)} \sum_{z \in B_u(y)} u(z), \\
 \varrho(x) &= \sum_{y \in \mathcal{P}(x)} \sum_{z \in B_u(y)} 1,
 \end{aligned} \tag{11.9}$$

where $\mathcal{P}(x)$ is a set of at most n_H centers of patches which are most similar to a patch centered at x . If the homogeneous region test is failed, the algorithm proceeds normally. Figure 11.5 shows some examples, computed for noisy images with different levels of noise. The points satisfying the homogeneity criterion are shown in white. With this trivial modification we can avoid usage of the similarity measure in the case it was not designed for.

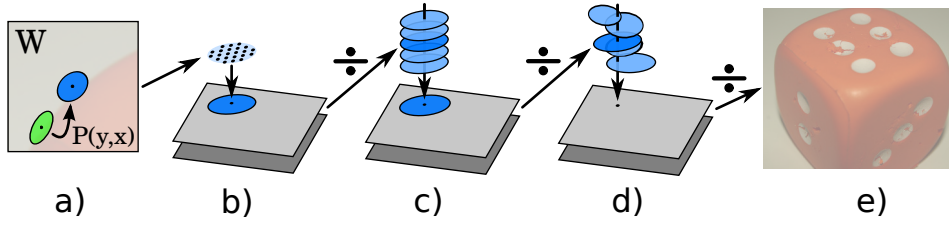


Figure 11.6: Diagram illustrating the aggregation step of the proposed method: a) normalization (W denotes the search window); b) interpolation; c) patch denoising; d) color estimates aggregation; e) resulting denoised image.

11.4 Implementation Details

As commented in Section 6.4, to compare elliptical patches in a more efficient way, we normalize them and interpolate to the regular grid. Since an input noisy image never changes during the denoising process, normalized patches at every point can be precomputed beforehand and stored together with their corresponding normalizing transformations. In the aggregation step, similar patches have to be transformed once again to match a reference patch. For that we calculate the complete transformation $P(y, x)$ using (9.8). Since halves of it, namely the normalizing transformations $R_u(x)T_u(x)^{\frac{1}{2}}$ and $R_u(y)T_u(y)^{\frac{1}{2}}$, are computed in advance and stored in memory, the computation of $P(y, x)$ takes only one matrix inversion and one matrix multiplication.

After transforming a similar patch to its reference patch, another interpolation is needed to aggregate color values from that similar patch. For this we use the same Nadaraya-Watson estimator as in the patch comparison, but for now the Gaussian standard deviation of the kernel is specified by a free parameter of the method, denoted by σ_{NW} . This parameter allows us to control to some extent the sharpness of the denoising result. The choice of its value, same for all levels of noise, is discussed in Section 11.5.

In the aggregation process, patches within the search window are weighted depending on their similarity (11.3) to the reference patch. This suggests the weight value of 1 for the reference patch itself. To avoid overweighting of the noise from the reference patch we instead take the maximum weight among the other patches in the search window.

For performance reason we use three pairs of buffers of the same size as an

input image during the denoising (see Figure 11.6). In every pair, the first buffer accumulates color values and the second buffer is for total weights. Normalized values are then obtained by element-wise division of the first buffer by the second. The first pair of buffers is used to interpolate every patch from within the search window after transforming it to its corresponding reference patch. The second pair of buffers is used to aggregate these transformed patches in order to denoise their reference patch. And, finally, the third pair of buffers aggregates color estimates for every pixel of the output image from different denoised patches.

11.5 Parameters Selection

The proposed method depends on the following parameters. Below, we explain the criteria to fix them.

- r – the “radius” used in the elliptical patch calculation (9.2). See Sections 5.1 and 9.1.
- ρ_{max} – the maximum size constraint for elliptical patches. It is specified in terms of the radius of a patch, shall it appear in a wide and completely homogeneous region. See Section 11.3.
- \hat{t} – the “scale” that controls the intra-patch Gaussian weights. See Sections 9.1.
- g – the resolution of the regular grid used for interpolation during the patch distance calculation. See Section 6.4.
- w – the size of the search window around a reference patch. See Section 11.3.
- b – the bandwidth multiplier. See Section 11.3.
- σ_{NW} – the interpolation coefficient used in the aggregation step. See Section 11.4.
- n_H – the number of most similar patches to be considered in the homogeneous region test. See Section 11.3.
- γ_H – the decision threshold of the homogeneous region test. See Section 11.3.

Table 11.1: Average PSNR values obtained for a test set of color images while varying the r and ρ_{max} with all the rest parameters being fixed. The best configuration is in **bold**

$\sigma = 2$				$\sigma = 5$			
r	25	30	35	ρ_{max}	15	20	25
ρ_{max}							
2	45.00	45.02	45.01	2	39.52	39.60	39.65
3	44.85	44.85	44.83	3	39.68	39.69	39.67
4	44.75	44.74	44.72	4	39.67	39.66	39.61

$\sigma = 10$				$\sigma = 20$			
r	20	25	30	ρ_{max}	40	45	50
ρ_{max}							
4	35.92	36.05	36.06	7	32.61	32.64	32.64
5	35.95	36.07	36.05	8	32.61	32.66	32.65
6	35.96	36.06	36.03	9	32.61	32.65	32.64

$\sigma = 30$				$\sigma = 40$			
r	60	65	70	ρ_{max}	85	90	95
ρ_{max}							
12	30.65	30.69	30.69	18	29.26	29.28	29.27
13	30.67	30.71	30.68	19	29.28	29.31	29.29
14	30.66	30.69	30.69	20	29.29	29.28	29.28

Table 11.2: Parameters of the method chosen for different noise levels σ

σ	ρ_{max}	r	w	g
2	2	30	29	9
5	3	20	29	9
10	5	25	31	9
20	8	45	33	13
30	13	65	35	13
40	19	90	35	21

Table 11.3: Parameters of the method that do not depend on the noise level

t	b	σ_{NW}	n_H	γ_H
1	0.35	0.4	30	0.35

Since it would be impractical to test all possible combinations of all the parameters, we split them into several groups and adjust one group at a time. We measure performance of the method on each configuration of

parameters by the average PSNR value, computed for the test set of eight color images shown in Figure 11.7.

The first group of parameters includes r , ρ_{max} and t which control the size of elliptical patches. Some of the PSNR values obtained while varying the r and ρ_{max} parameters are shown in Table 11.1 as an example. Let us remark that only a few values around the maximum are shown for every level of noise; however, much wider ranges were actually tested.

The second group of parameters includes w and g which as well should be picked for every noise level separately. The size of the search window w should be bigger for higher levels of noise to provide more patch candidates. The value of g should roughly follow the value of ρ_{max} to ensure enough resolution of the regular grid to represent normalized patches. These intuitions are confirmed while testing the ranges of values for these two parameters.

The third group of parameters includes b and σ_{NW} . In contrast to the original work of Buades et al. (2005) we have not observed any significant effect of varying the b parameter for different noise levels. The value of σ_{NW} has no relations with the noise level; therefore, a single value was picked for all the experiments.

The fourth group of parameters includes n_H and γ_H which are associated with the homogeneous region test. The same values were picked for all levels of noise.

The values of parameters selected for different levels of noise are summarized in Table 11.2. The values of parameters, that do not depend on the noise level and thus are fixed for all our experiments, are summarized in Table 11.3.

11.6 Experimental Results and Assessment

In this section we present assessment of the proposed method and compare it with the original Non-Local Means method of Buades et al. (2005) and the state-of-the-art Non-Local Bayes method of Lebrun et al. (2013b). In order to assess the performance of the proposed method, we demonstrate the “method noise” and the “noise to noise” benchmarks commonly used in the literature. For a quantitative evaluation, we show the PSNR values computed for multiple test images and different values of the standard deviation of noise. Finally, we show some of the denoised images for a qualitative assessment.

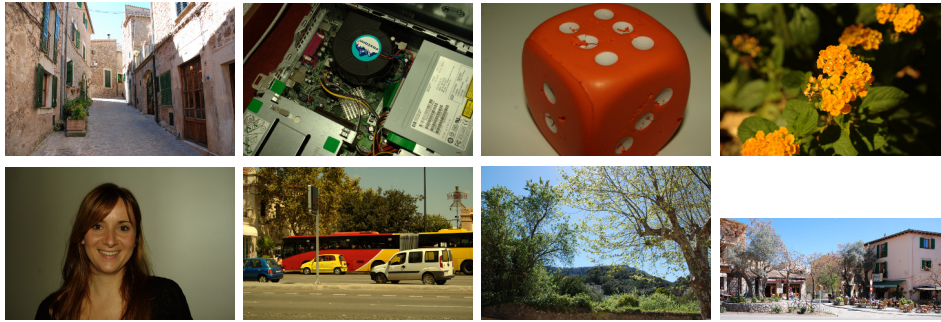


Figure 11.7: Noiseless images used in the experiments: Alley, Computer, Dice, Flowers, Girl, Traffic, Trees and Valldemossa.

To obtain the results for Non-Local Means and Non-Local Bayes, we use public implementations made by the authors (Buades et al. (2011); Lebrun et al. (2013a)) and available online at <http://www.ipol.im/>. For PSNR evaluation, visual comparison and some other experiments we use the set of eight color images shown in Figure 11.7, available online at <http://www.ipol.im/> under a Creative Commons CC-BY license. Images Alley and Valldemossa by A. Buades, the rest six images by M. Colom.

“Method Noise” and “Noise to Noise” Assessments

At first, we assess the proposed method by calculating the so called “method noise” (Buades et al. (2005)). For that, we calculate the difference between a noisy image and an output of a denoising method. The “method noise” should contain as little structure from a noisy image as possible. Figure 11.8 shows the comparison between the Non-Local Bayes, the original Non-Local Means and our proposed method. It is easy to see that for the proposed method the “method noise” looks almost like a white noise and is very close to the one of the Non-Local Bayes method.

According to the “noise to noise” principle (Lebrun et al. (2012)), a denoising method should transform white noise into white noise. If in contrast a method creates some structure from that noise, it will introduce similar artifacts in the denoising result. For this benchmark a uniform image with color values (127, 128, 129) is used as an input. White noise of standard deviation 30 is added separately to every channel of that image and the obtained noisy image is then processed by a denoising method. For visualization purposes

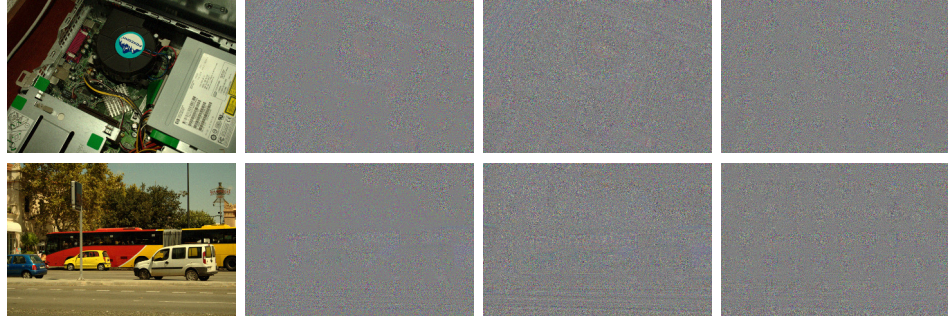


Figure 11.8: “Method noise” benchmark. From left to right: noisy image with added Gaussian white noise of standard deviation $\sigma = 5$, “method noise” for the Non-Local Bayes method, for the original Non-Local Means method and for our method. For the visualization purposes, the difference values were scaled from the range of $[-4\sigma, 4\sigma]$ to the range of $[0, 255]$.

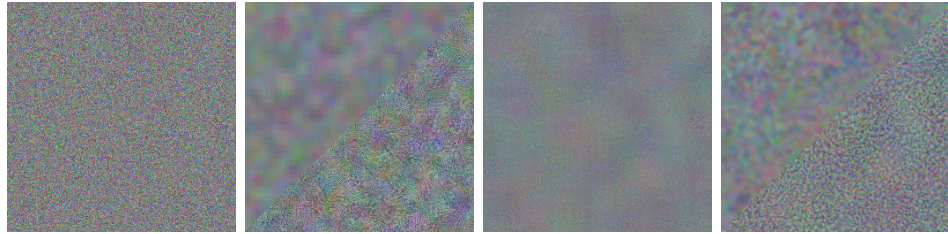


Figure 11.9: “Noise to noise” benchmark. From left to right: an image of uniform color (127, 128, 129) with added Gaussian noise of standard deviation $\sigma = 30$, results of applying the Non-Local Bayes, the original Non-Local Means and our proposed method. Note that for the Non-Local Bayes and for our method, top-left half was processed as homogeneous region (as it should be) while bottom-right as non-homogeneous (this might happen if the homogeneity criterion is not accurate).

the resulting denoised images are amplified: for every pixel the difference to the mean color (128, 128, 128) is magnified by the factor of 5 (see [Lebrun et al. \(2012\)](#) for the complete methodology). Figure 11.9 shows the “noise to noise” benchmark for the Non-Local Bayes, the original Non-Local Means and our proposed method. Notice that for the Non-Local Bayes and for our method the result of this benchmark depends on whether the noisy image is treated as homogeneous or as non-homogeneous region. We show both outcomes in a single image, where top-left and bottom-right halves correspond,

respectively, to homogeneous and non-homogeneous branches of the algorithms. As can be seen, the proposed method leaves no structural artifacts in both cases. In contrast, the Non-Local Bayes method introduces some artifacts when the noisy image is processed as non-homogeneous region. This indicates that the homogeneity threshold γ of that method should be carefully adjusted to avoid such artifacts in real images.

PSNR and Visual Comparison

Mean square error (MSE) and peak signal-to-noise ratio (PSNR) are commonly used in the denoising literature to quantify a relative performance of different denoising methods. Tables 11.4 and 11.5 show PSNR values for denoising results obtained for the eight color images of Figure 11.7 and standard deviation of noise $\sigma \in \{2, 5, 10, 20, 30, 40\}$. As expected, our method almost always outperforms the original Non-Local Means method in terms of PSNR. For high levels of noise, PSNR values for our method are not very far from the ones of the state-of-the-art Non-Local Bayes method.

Even though comparison by PSNR provides a handy objective metric, an important evaluation is visual comparison that for now can only be done by a human being. Figure 11.10 shows several denoising results for high levels of noise. To better illustrate the comparison, Figure 11.11 shows the same results zoomed around some interesting regions. As can be seen, the original Non-Local Means suppresses small image details and thus produces “flattened” results. This is especially noticeable in the images Alley, Girl and Flowers. The Non-Local Bayes, while providing high PSNR values, may introduce the so-called staircasing effect – abrupt jumps in colors in regions where a smooth transition should take place. Appearance of these new edges leads to a mosaicing that is most visible in the images Dice, Girl and Flowers. It is also affected by the noise halo (*rare patch effect*, Deledalle et al. (2012)) around high contrasted edges which can be observed in the images Traffic and Computer. Our method is capable of denoising both small image details, sharp edges and smooth color transitions. It can be noticed, however, that in some particular cases the proposed method may oversmooth or suppress small image details. Another problem is a low frequency noise that remains in homogeneous regions and can be explained by the effect that noise has on the shape adaptive patches. Figure 11.12 illustrates some failure cases that actually can be observed in the denoising results of all methods being compared at high level of noise.

Table 11.4: PSNR values for noise $\sigma = 2, 5$ and 10

$\sigma = 2$			
	NL-Bayes	NL-means	our
Alley	45.28	42.68	43.37
Computer	45.81	43.93	44.67
Dice	49.17	48.12	48.22
Flowers	47.75	46.31	46.89
Girl	47.67	46.71	46.71
Traffic	45.17	43.45	44.00
Trees	43.44	42.15	42.62
Valldemossa	45.07	43.26	43.67

$\sigma = 5$			
	NL-Bayes	NL-means	our
Alley	39.14	37.25	37.31
Computer	40.54	38.93	39.15
Dice	46.02	44.93	45.22
Flowers	43.29	42.17	42.74
Girl	44.18	43.36	43.37
Traffic	39.39	37.59	38.01
Trees	36.54	34.71	35.03
Valldemossa	38.62	35.96	36.70

$\sigma = 10$			
	NL-Bayes	NL-means	our
Alley	34.82	33.64	33.55
Computer	36.68	35.54	35.40
Dice	43.20	41.92	42.63
Flowers	39.53	38.59	39.38
Girl	41.43	40.40	40.85
Traffic	35.15	34.05	34.10
Trees	31.70	29.59	30.31
Valldemossa	33.96	32.15	32.33

11.7 Conclusions

In this chapter we have presented an extension for the Non-Local Means image denoising method that considers shape-adaptive patches instead of the very common square ones. The affine invariant patch similarity measure (9.9) allows us to compare patches related by an affinity and thus plays an

Table 11.5: PSNR values for noise $\sigma = 20, 30$ and 40

$\sigma = 20$			
	NL-Bayes	NL-means	our
Alley	31.17	29.98	30.15
Computer	32.98	31.67	32.03
Dice	40.17	38.31	39.62
Flowers	36.14	34.53	36.08
Girl	38.62	36.92	38.02
Traffic	31.24	30.14	30.46
Trees	27.36	26.37	26.40
Valldemossa	29.72	28.44	28.51

$\sigma = 30$			
	NL-Bayes	NL-means	our
Alley	29.15	27.85	28.37
Computer	30.68	29.28	30.06
Dice	37.95	36.92	37.54
Flowers	33.85	32.35	34.11
Girl	36.69	35.58	36.25
Traffic	29.03	27.74	28.54
Trees	25.03	23.79	24.31
Valldemossa	27.35	25.89	26.46

$\sigma = 40$			
	NL-Bayes	NL-means	our
Alley	27.77	26.48	27.11
Computer	29.04	27.61	28.57
Dice	36.24	35.26	36.05
Flowers	32.13	30.58	32.70
Girl	35.06	34.17	35.00
Traffic	27.52	26.23	27.17
Trees	23.50	22.42	22.88
Valldemossa	25.81	24.46	25.04

essential role in our method. By extending the space of patches being considered, this similarity measure helps to detect larger amount of similar noisy patches. This in turn leads to better denoising results. Moreover, shape-adaptive patches are well-suited for denoising along contrasted edges, as a result, we obtain denoised images that does not suffer from the halo artifacts.

After introducing the denoising problem itself and the original Non-Local Means method, we have described the proposed extension and have suggested a slight modification of the similarity measure (9.9) that better adapts it for denoising application. In addition, we have introduced an idea of separating the treatment of homogeneous and textured regions, that was not exploited in the original Non-Local Means. To facilitate reproduction of the presented results we have commented on the most important implementation details of the algorithm and have also specified values of parameters being used in the experiments. We have conducted the assessment of our extension by the so-called “method noise” and “noise to noise” benchmarks from the literature. Finally, we have shown that our extension outperforms the original Non-Local Means method both quantitatively and qualitatively. Furthermore, while showing slightly smaller PSNR values than the state-of-the-art Non-Local Bayes method, the proposed method provides results of comparable or even better visual quality. We find these observations to be promising and consider a similar extension of Non-Local Bayes method as a possible direction for future research. It will also be useful to perform a focus group evaluation of the denoising results produced by different methods. Another subject for future work is the low frequency noise that remains in homogeneous regions for the noise levels of $\sigma = 30$ and higher. Perhaps a better suppression of this residual noise can be achieved using multiscale and two-step approaches to denoising.

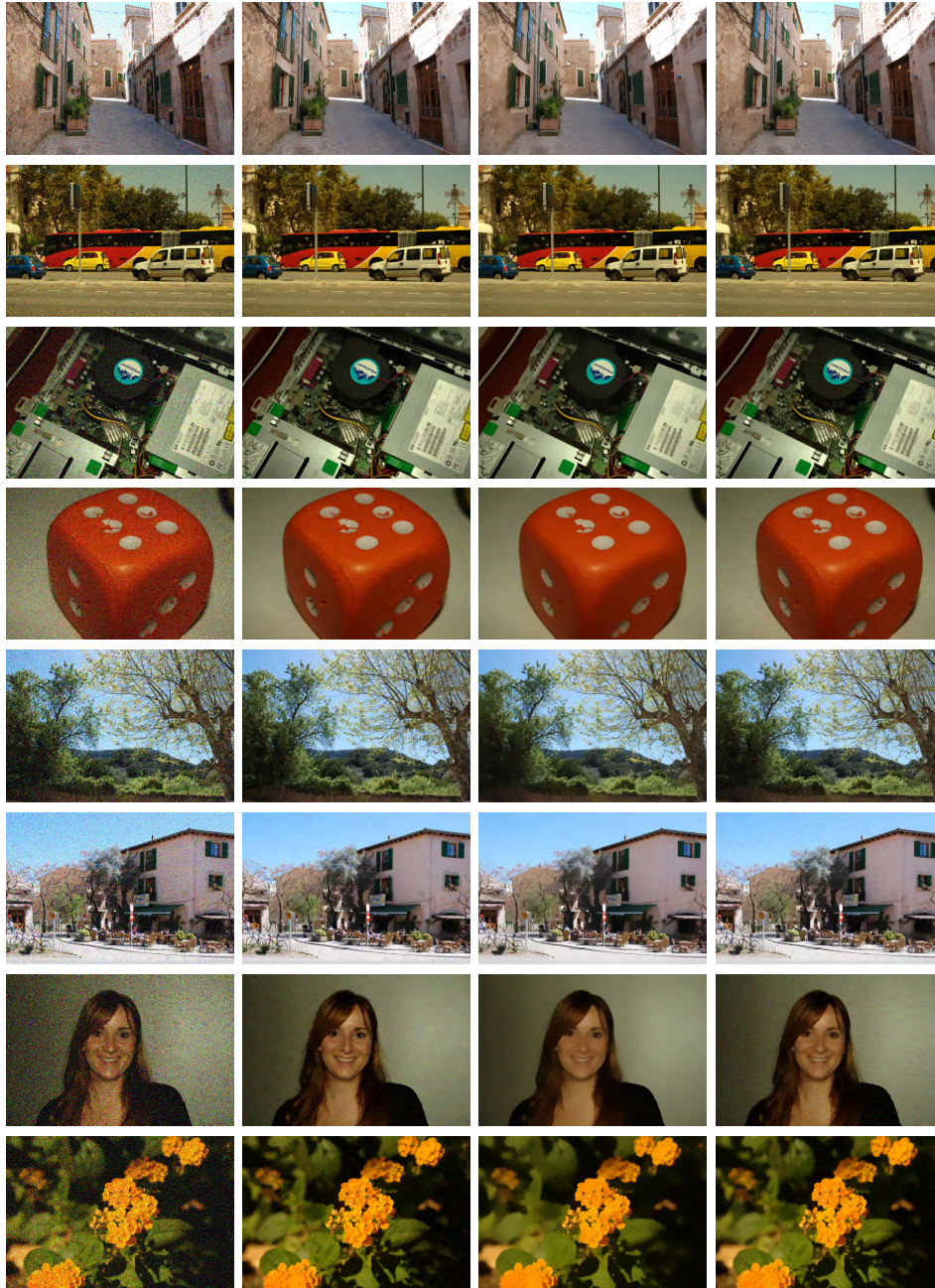


Figure 11.10: Denoising results. In columns: noisy image, NL-Bayes, NL-Means, our method. In rows: Alley (noise $\sigma = 20$), Traffic (noise $\sigma = 20$), Computer (noise $\sigma = 30$), Dice (noise $\sigma = 30$), Trees (noise $\sigma = 30$), Valldemossa (noise $\sigma = 30$), Girl (noise $\sigma = 40$), Flowers (noise $\sigma = 40$).

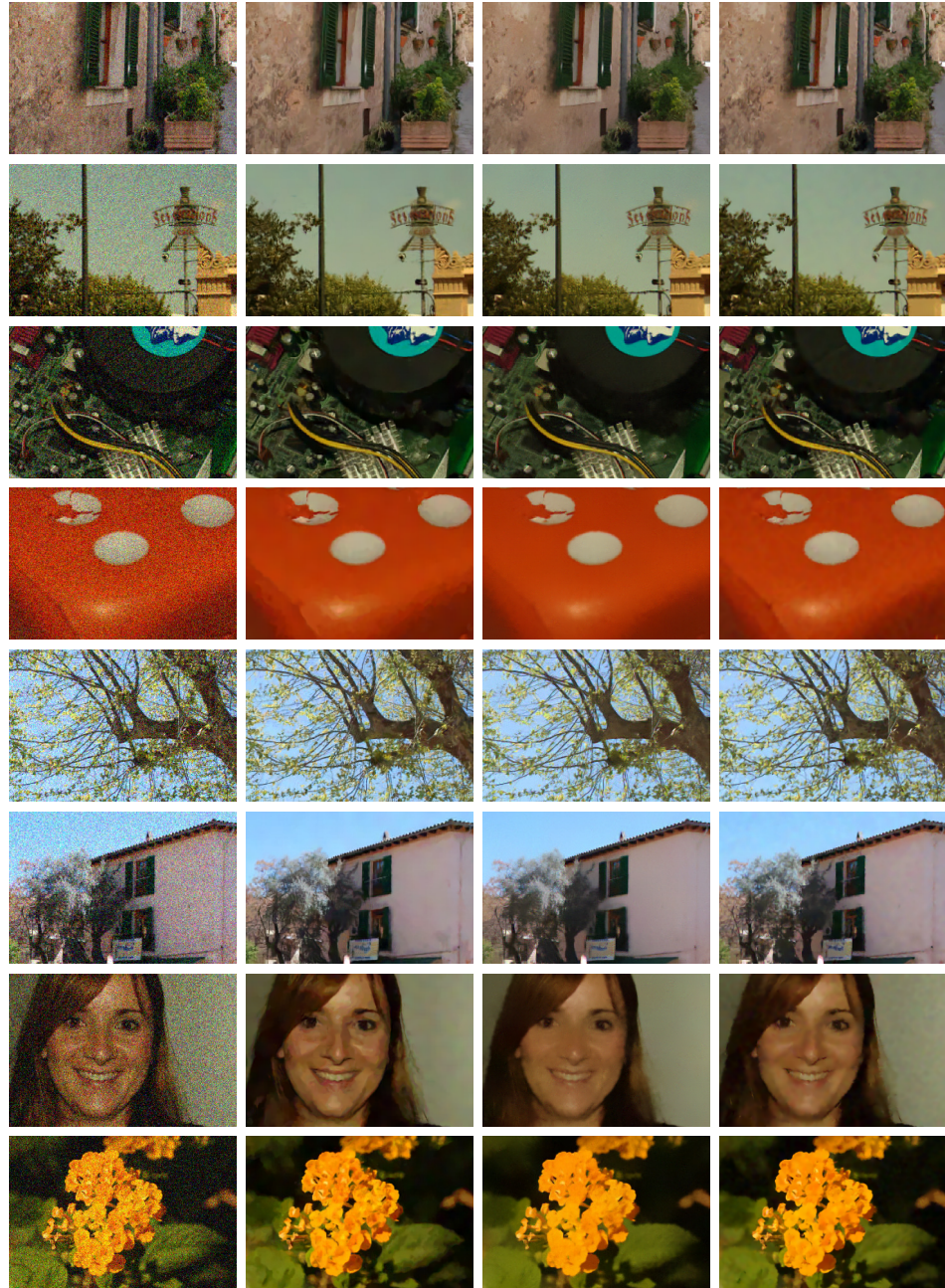


Figure 11.11: Zoomed denoising results. In columns: noisy image, NL-Bayes, NL-Means, our method. In rows: Alley (noise $\sigma = 20$), Traffic (noise $\sigma = 20$), Computer (noise $\sigma = 30$), Dice (noise $\sigma = 30$), Trees (noise $\sigma = 30$), Valldemossa (noise $\sigma = 30$), Girl (noise $\sigma = 40$), Flowers (noise $\sigma = 40$).

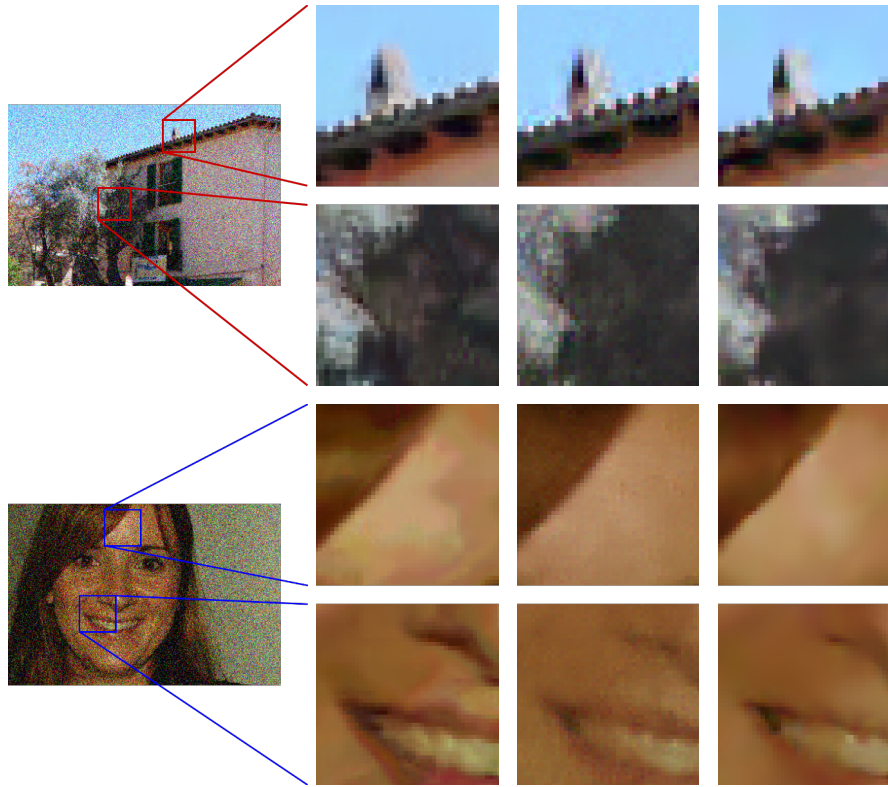


Figure 11.12: Closeups of some failure cases in denoised images Valldemossa (noise $\sigma = 30$) and Girl (noise $\sigma = 40$). In columns: noisy image, NL-Bayes, NL-Means, our method. In rows: rare patch effect in NL-Bayes and NL-Means results, occasional oversmoothing of details in NL-Means and our results, staircasing effect in NL-Bayes result, suppression of details in NL-Means and our results.

Conclusions

In the second part of the thesis we have presented two novel image processing methods for image inpainting and image denoising that exploit the affine invariant patch similarity measure (4.17) proposed and studied in Part I. These promising methods should be considered as two major contributions of the second part of the thesis. We have started off with a revision of our approach to image comparison. In order to provide a better intuition, we have shown a way to *engineer* the same similarity measure (9.9) from bottom up, regardless the theoretical framework of multiscale analyses of similarity measures.

In the context of exemplar-based image inpainting the affine invariant similarity measure allows us to perform completion of a texture under perspective effect and lens distortion, and also to inpaint one view of a scene, using information from another view as a source. We have suggested a variational formulation of the inpainting problem with the similarity measure (9.9) embedded in it. From that formulation we have obtained an efficient approximate minimization scheme that exploits the affine covariance property of the structure tensors to directly compute an appropriate transformation between a pair of shape-adaptive patches. We have discussed the numerical implementation of that minimization scheme and have also illustrated the capabilities and weaknesses of the proposed affine invariant image inpainting method by a set of experiments.

Finally, we have extended the well-known Non-Local Means image denoising method. The affine invariant similarity measure allows us to detect and aggregate more similar noisy patches which in turn leads to a better denoising performance. We have suggested a modification of the similarity

measure (9.9) that better adapts it for denoising problem. As usual, we have discussed the most important details of the numerical implementation of the proposed extension. Lastly, we have evaluated our method quantitatively, by comparing it in terms of PSNR values with the original Non-Local Means method and the state-of-the-art Non-Local Bayes method, and qualitatively, by the “method noise” and “noise to noise” benchmarks and also by the visual comparison of the denoising results.

We have covered only two possible applications of the affine invariant similarity measures studied in this work. Both inpainting and denoising methods have potential for further improvements; moreover, these similarity measures might prove useful in many other image and video processing problems, for instance, in stereo vision, segmentation, texture replacement, registration, etc.

PART III

Appendices

Proof of Theorem 4.1

Lemma A.1. *Let \mathcal{M} be a Riemannian manifold. Let D be a matrix such that*

$$RDR^t = D$$

for all rotations R in $(T_\xi(M), G(\xi))$. Then $D = \lambda G(\xi)^{-1}$ for some $\lambda \in \mathbb{R}$.

The proof of this Lemma can be found in [Calderero and Caselles \(2014\)](#) (Lemma 5.1).

Theorem A.2. *Let T_t be a multiscale analysis of similarity functions satisfying all Architectural axioms, the Comparison principle, and Gray level Shift invariance axiom. Assume also that T_t is linear. Then*

$$\frac{\partial C}{\partial t} = F(D_{\mathcal{N}}^2 C, \xi, G),$$

where

$$\begin{aligned} F(X, \xi, G) = & c_{11}(\xi) \text{Tr}((G^1)^{-1}(\xi_1) X_{11}) + 2c_{12}(\xi, G) \text{Tr}(\bar{D}_{12} I^1(\xi_1)^{-1} X_{12}) \\ & + c_{22}(\xi) \text{Tr}((G^2)^{-1}(\xi_2) X_{22}), \end{aligned}$$

and \bar{D}_{12} is an isometry from $(T_{\xi_1} \mathcal{M}^1, G^1(\xi_1)) \rightarrow (T_{\xi_2} \mathcal{M}^2, G^2(\xi_2))$. The ellipticity of F implies that $c_{11}, c_{22} \geq 0$.

Proof. For any symmetric matrix $X = [X_{ij}] \in SM_\xi(\mathcal{N})$, any $p \in T_\xi^* \mathcal{N}$, and $c \in \mathbb{R}$, let the function \tilde{F} be defined by the identity

$$\tilde{F}(X, p, c, \xi, G, \Gamma^k) = F(X + \Gamma(p), p, c, \xi, G, \Gamma^k).$$

Since T_t is gray level shift invariant, then \tilde{F} does not depend on c . On the other hand, it does not depend on Γ^k . The linearity of T_t and Theorem 3.1 imply that in terms of the function \tilde{F}

$$\tilde{F}(rX_1 + sX_2, rp_1 + sp_2, \xi, G) = r\tilde{F}(X_1, p_1, \xi, G) + s\tilde{F}(X_2, p_2, \xi, G)$$

for any $X_1, X_2 \in \text{SM}_\xi(\mathcal{N})$, any $p_1, p_2 \in T_\xi^*\mathcal{N}$, any $r, s \in \mathbb{R}$. By taking $X_1 = X$, $X_2 = 0$, $p_1 = 0$, $p_2 = p$, $r = 1$, $s = 1$, we write

$$\tilde{F}(X, p, \xi, G) = \tilde{F}(X, 0, \xi, G) + \tilde{F}(0, p, \xi, G) =: K'(X, \xi, G) + K''(p, \xi, G), \quad (\text{A.1})$$

where K' is linear in X and K'' is linear in p . Moreover, from the rotation invariance of \tilde{F}

$$\tilde{F}(X, p, \xi, G) = \tilde{F}(R^t X R, R^t p, \xi, G),$$

for all diagonal rotations R in $(T_\xi \mathcal{N}, G(\xi))$. Recall that in this context “diagonal rotation” means $R = (R_1, R_2)$, where R_1 and R_2 are related. We deduce that

$$K'(X, \xi, G) = K'(R^t X R, \xi, G) \quad (\text{A.2})$$

$$K''(p, \xi, G) = K''(R^t p, \xi, G). \quad (\text{A.3})$$

Let us write

$$X = \begin{bmatrix} X_{11} & X_{12} \\ X_{21} & X_{22} \end{bmatrix},$$

where $X_{21} = X_{12}^t$. We write also

$$R = \begin{bmatrix} R_1 & 0 \\ 0 & R_2 \end{bmatrix},$$

where $R_1 : (T_{\xi_1} \mathcal{M}^1, G^1(\xi_1)) \rightarrow (T_{\xi_1} \mathcal{M}^1, G^1(\xi_1))$, and $R_2 : (T_{\xi_2} \mathcal{M}^2, G^2(\xi_2)) \rightarrow (T_{\xi_1} \mathcal{M}^2, G^2(\xi_2))$, with $R_2 \bar{P}(\xi) = P(\xi) R_1$. Then

$$R^t X R = \begin{bmatrix} R_1^t X_{11} R_1 & R_1^t X_{12} R_2 \\ R_2^t X_{21} R_1 & R_2^t X_{22} R_2 \end{bmatrix}.$$

At first we concentrate on (A.2). Since K' is linear in X , we have

$$K'(X, \xi, G) = K'_{11}(X_{11}, \xi, G) + K'_{12}(X_{12}, \xi, G) + K'_{21}(X_{21}, \xi, G) + K'_{22}(X_{22}, \xi, G),$$

where each $K'_{ij}(X_{ij}, \xi, G)$ is linear in X_{ij} and by (A.2) we have

$$K'_{11}(R_1^t X_{11} R_1, \xi, G) = K'_{11}(X_{11}, \xi, G),$$

$$K'_{12}(R_1^t X_{12} R_2, \xi, G) + K'_{21}(R_2^t X_{21} R_1, \xi, G) = K'_{12}(X_{12}, \xi, G) + K'_{21}(X_{21}, \xi, G), \quad (\text{A.4})$$

$$K'_{22}(R_2^t X_{22} R_2, \xi, G) = K'_{22}(X_{22}, \xi, G).$$

At ξ fixed, and for $i = 1, 2$, $K'_{ii}(X_{ii}, \xi, G)$ is a symmetric linear function of the eigenvalues of X_{ii} , then there exists a matrix $D_{ii} : (T_{\xi_i} \mathcal{M}^i, G^i(\xi_i)) \rightarrow (T_{\xi_i} \mathcal{M}^i, G^i(\xi_i))$ (depending on ξ, G) such that

$$K'_{ii}(X_{ii}, \xi, G) = \text{Trace}(D_{ii} X_{ii}).$$

From the rotation invariance

$$\text{Trace}(D_{ii} X_{ii}) = \text{Trace}(D_{ii} R_i^t X_{ii} R_i) = \text{Trace}(R_i D_{ii} R_i^t X_{ii}).$$

Since this is true for all X_{ii} , then $R_i D_{ii} R_i^t = D_{ii}$. By Lemma A.1 we have that $D_{ii} = c_{ii}(\xi, G)(G^i)^{-1}(\xi_i)$ for some constant $c_{ii}(\xi, G)$.

Therefore, we can rewrite (A.1) as

$$\begin{aligned} \tilde{F}(X, p, \xi, G) &= c_{11}(\xi, G) \text{Tr}((G^1)^{-1}(\xi_1) X_{11}) + c_{22}(\xi, G) \text{Tr}((G^2)^{-1}(\xi_2) X_{22}) \\ &\quad + K'_{12}(X_{12}, \xi, G) + K'_{21}(X_{21}, \xi, G) + K''(p, \xi, G), \end{aligned}$$

Now, $K'_{12}(X_{12}, \xi, G) + K'_{21}(X_{21}, \xi, G)$ is a linear function of X_{12} . Thus, there is a matrix D'_{ij} such that

$$K'_{12}(X_{12}, \xi, G) + K'_{21}(X_{21}, \xi, G) = \text{Trace}(D'_{12} X_{12}).$$

Since the map has to be an endomorphism, $D'_{12} X_{12}$ has to be a map $(T_{\xi_2} \mathcal{M}^2, G^2(\xi_2)) \rightarrow (T_{\xi_2} \mathcal{M}^2, G^2(\xi_2))$. However, $X_{12} : (T_{\xi_2} \mathcal{M}^2, G^2(\xi_2)) \rightarrow (T_{\xi_1} \mathcal{M}^1, G^1(\xi_1))^*$; therefore, we write the map $D'_{12} = D_{12} I(\xi_1)^{-1}$, where $I(\xi_1) : (T_{\xi_1} \mathcal{M}^1, G^1(\xi_1)) \rightarrow (T_{\xi_1} \mathcal{M}^1, G^1(\xi_1))^*$ and $D_{12} : (T_{\xi_1} \mathcal{M}^1, G(\xi_1)^1) \rightarrow (T_{\xi_2} \mathcal{M}^2, G^2(\xi_2))$. Using the rotation invariance (A.4) we have

$$\begin{aligned} \text{Trace}(D_{12} I^1(\xi_1)^{-1} X_{12}) &= \text{Trace}(D_{12} I^1(\xi_1)^{-1} R_1^t X_{12} R_2) \quad (\text{A.5}) \\ &= \text{Trace}(R_2 D_{12} I^1(\xi_1)^{-1} R_1^t X_{12}). \end{aligned}$$

Note that all maps inside the traces map $(T_{\xi_2} \mathcal{M}^2, G^2(\xi_2)) \rightarrow (T_{\xi_2} \mathcal{M}^2, G^2(\xi_2))$. This implies that

$$D_{12} I(\xi_1)^{-1} = R_2 D_{12} I(\xi_1)^{-1} R_1^t,$$

that is,

$$D_{12} = R_2 D_{12} I^1(\xi_1)^{-1} R_1^t I^1(\xi_1),$$

as maps from $(T_{\xi_1}\mathcal{M}^1, G^1(\xi_1)) \rightarrow (T_{\xi_2}\mathcal{M}^2, G^2(\xi_2))$. Let us observe that $I^1(\xi_1)^{-1}R_1^t I^1(\xi_1) : (T_{\xi_1}\mathcal{M}^1, G^1(\xi_1)) \rightarrow (T_{\xi_1}\mathcal{M}^1, G^1(\xi_1))$ is an isometry and denote it by $\bar{R}_1^t = I^1(\xi_1)^{-1}R_1^t I^1(\xi_1)$. We can write

$$D_{12}\bar{R}_1^{-t} = R_2 D_{12}. \quad (\text{A.6})$$

We interpret R_2 as a representation of the isometry group of $(T_{\xi_2}\mathcal{M}^2, G^2(\xi_2))$, and \bar{R}_1^{-t} is a representation of the isometry group of $(T_{\xi_1}\mathcal{M}^1, G^1(\xi_1))$. Denote them by $\bar{\rho}(\mathcal{R})$ and $\rho(\mathcal{R})$, respectively (\mathcal{R} represents a rotation). Notice the slight abuse of notation writing \mathcal{R} in both cases, but note that R_1 is determined by R_2 , since both are $P(\xi)$ -related. Then $\rho(I) = I$ and $\bar{\rho}(I) = I$. Note also that $I = \rho(\mathcal{R}\mathcal{R}^{-1}) = \rho(\mathcal{R})\rho(\mathcal{R}^{-1})$; thus, $\rho(\mathcal{R}^{-1}) = \rho(\mathcal{R})^t$ (they are isometries). Similarly for $\bar{\rho}(\mathcal{R})$. We rewrite (A.6) as

$$D_{12}\rho(\mathcal{R}) = \bar{\rho}(\mathcal{R})D_{12} \quad \forall \mathcal{R}. \quad (\text{A.7})$$

By transposing we have

$$\rho(\mathcal{R})^t D_{12}^t = D_{12}^t \bar{\rho}(\mathcal{R})^t \quad \forall \mathcal{R}. \quad (\text{A.8})$$

After multiplying by D_{12} ,

$$D_{12}\rho(\mathcal{R})^t D_{12}^t = D_{12}D_{12}^t \bar{\rho}(\mathcal{R})^t \quad \forall \mathcal{R}. \quad (\text{A.9})$$

Writing (A.7) with \mathcal{R}^{-1} instead of \mathcal{R} and using the fact that $\rho(\mathcal{R}^{-1}) = \rho(\mathcal{R})^t$ and $\bar{\rho}(\mathcal{R}^{-1}) = \bar{\rho}(\mathcal{R})^t$, we have

$$D_{12}\rho(\mathcal{R})^t = \bar{\rho}(\mathcal{R})^t D_{12} \quad \forall \mathcal{R}. \quad (\text{A.10})$$

Combining (A.10) with (A.9), we have

$$\bar{\rho}(\mathcal{R})^t D_{12}D_{12}^t = D_{12}D_{12}^t \bar{\rho}(\mathcal{R})^t \quad \forall \mathcal{R}. \quad (\text{A.11})$$

Then, by Schur's Lemma (see Kanatani (1990)), there is a constant $c \in \mathbb{R}$ such that $D_{12}D_{12}^t = cI$. Thus, either $D_{12} = 0$ or $\frac{1}{\sqrt{c}}D_{12}$ is an isometry in $(T_{\xi_1}\mathcal{M}^1, G^1(\xi_1)) \rightarrow (T_{\xi_2}\mathcal{M}^2, G^2(\xi_2))$. Note that this cannot be improved since reading this backwards, we have that (A.4) holds. We denote this constant c as $2c_{12}(\xi, G)$.

We have proved that

$$K'_{12}(X_{12}, \xi, G) + K'_{21}(X_{21}, \xi, G) = 2c_{12}(\xi, G)\text{Trace}(\bar{D}_{12}I^1(\xi_1)^{-1}X_{12}), \quad (\text{A.12})$$

where $\bar{D}_{12} : (T_{\xi_1} \mathcal{M}^1, G^1(\xi_1)) \rightarrow (T_{\xi_2} \mathcal{M}^2, G^2(\xi_2))$ is an isometry.

Since for $i = 1, 2$, we can write

$$\begin{aligned} K'_{ii}(X_{ii}, \xi, G) &= c_{ii}(\xi, G) \text{Trace}((G^i)^{-1}(\xi_i) X_{ii}) \\ &= c_{ii}(\xi, G) \text{Trace}(B^i(\xi_i)(B^i)^t(\xi_i) X_{ii}) \\ &= c_{ii}(\xi, G) \text{Trace}((B^i)^t(\xi_i) X_{ii} B^i(\xi_i)) \\ &= \mathcal{H}_{ii}(B^i(\xi_i)^t X_{ii} B^i(\xi_i), 0, \xi), \end{aligned}$$

where \mathcal{H}_{ii} is linear in its first argument (see Proposition 3.6 in Ballester et al. (2014)), we deduce that $c_{ii}(\xi, G)$ does not depend on G . The ellipticity of \tilde{F} proves that $c_{ii}(\xi) \geq 0$, $i = 1, 2$.

We can rewrite (A.1) as

$$\begin{aligned} \tilde{F}(X, p, \xi, G) &= c_{11}(\xi) \text{Tr}((G^1)^{-1}(\xi_1) X_{11}) + 2c_{12}(\xi, G) \text{Tr}(\bar{D}_{12} I^1(\xi_1)^{-1} X_{12}) \\ &\quad + c_{22}(\xi) \text{Tr}((G^2)^{-1}(\xi_2) X_{22}) + K''(p, \xi, G), \end{aligned}$$

Let us now prove that $K''(p, \xi, G) = 0$. By (A.3) we have

$$K''(p, \xi, G) = K''(R^t p, \xi, G),$$

for all diagonal rotations as above. Let $p = (p_1, p_2)$. Then we may write

$$K''(p, \xi, G) = K''_1(p_1, \xi, G) + K''_2(p_2, \xi, G),$$

where K''_i is linear in p_i (ξ, G fixed). Thus, letting $p_1 = 0$ and $p_2 = 0$, respectively, we deduce

$$K''_i(p_i, \xi, G) = K''_i(R_i^t p_i, \xi, G)$$

for $i = 1, 2$. Thus, K''_i does not depend on p_i , only on its modulus, that is

$$K''_i(p_i, \xi, G) = \bar{K}''_i(|p_i|_{(G^i)^{-1}}, \xi, G)$$

for some function \bar{K}''_i .

Let us compute the modulus. Observe that

$$\langle R_i^t p_i, R_i^t p'_i \rangle = ((G^i)^{-1}(\xi_i) R_i^t p_i, R_i^t p'_i) = (R_i (G^i)^{-1}(\xi_i) R_i^t p_i, p'_i).$$

From $R_i^t G^i(\xi_i) R_i = G^i(\xi_i)$, we have $R_i (G^i)^{-1}(\xi_i) R_i^t = (G^i)^{-1}(\xi_i)$. Thus,

$$(R_i (G^i)^{-1}(\xi_i) R_i^t p_i, p'_i) = ((G^i)^{-1}(\xi_i) p_i, p'_i) = \langle p_i, p'_i \rangle.$$

Thus, $|R_i^t p_i|_{(g^i)^{-1}(\xi_i)} = |p_i|_{(g_i)^{-1}(\xi_i)}$ for any covector p_i . Then

$$2\bar{K}_i''(|p_i|_{(g^i)^{-1}(\xi_i)}, \xi, G) = K_i''(p_i, \xi, G) + K_i''(-p, \xi, G) = K_i''(0, \xi, G) = 0.$$

Finally, we can rewrite (A.1) as

$$\begin{aligned} \tilde{F}(X, \xi, G) = & c_{11}(\xi) \text{Tr}((G^1)^{-1}(\xi_1) X_{11}) + 2c_{12}(\xi, G) \text{Tr}(\bar{D}_{12} I^1(\xi_1)^{-1} X_{12}) \\ & + c_{22}(\xi) \text{Tr}((G^2)^{-1}(\xi_2) X_{22}). \end{aligned}$$

Therefore, our claim is proved. \square

Bibliography

Each reference indicates the pages where it appears.

- A. Almansa, V. Caselles, G. Haro, and B. Rougé. Restoration and zoom of irregularly sampled, blurred, and noisy images by accurate total variation minimization with local constraints. *Multiscale Modeling and Simulation*, 5(1):235–72, 2006.
- L. Alvarez, P.L. Lions, and J.-M. Morel. Image selective smoothing and edge detection by nonlinear diffusion. ii. *SIAM Journal on numerical analysis*, 29(3):845–866, 1992.
- L. Alvarez, F. Guichard, P.L. Lions, and J.M. Morel. Axioms and fundamental equations of image processing. *Archive for rational mechanics and analysis*, 123(3):199–257, 1993.
- P. Arias, G. Facciolo, V. Caselles, and G. Sapiro. A variational framework for exemplar-based image inpainting. *International Journal of Computer Vision*, 93:319–347, 2011.
- G. Aubert and J. F. Aujol. A variational approach to removing multiplicative noise. *SIAM Journal on Applied Mathematics*, 68(4):925–946, 2008.
- J.-F. Aujol, S. Ladjal, and S. Masnou. Exemplar-based inpainting from a variational point of view. *SIAM Journal on Mathematical Analysis*, 42(3):1246–1285, 2010.
- C. Ballester and M. Gonzalez. Affine invariant texture segmentation and shape from texture by variational methods. *Journal of Mathematical Imaging and Vision*, 9(2):141–171, 1998.
- C. Ballester, M. Bertalmío, V. Caselles, G. Sapiro, and J. Verdera. Filling-in

- by joint interpolation of vector fields and gray levels. *IEEE Transactions on Image Processing*, 10(8):1200–1211, 2001.
- C. Ballester, F. Calderero, V. Caselles, and G. Facciolo. Multiscale analysis of similarities between images on riemannian manifolds. *SIAM Journal Multiscale Modeling and Simulation*, 12(2):616–649, 2014.
- C. Barnes, E. Shechtman, A. Finkelstein, and D. B. Goldman. PatchMatch: a randomized correspondence algorithm for structural image editing. In *ACM SIGGRAPH 2009 Papers*, pages 1–11. ACM, 2009.
- C. Barnes, E. Shechtman, D. B. Goldman, and A. Finkelstein. The generalized PatchMatch correspondence algorithm. In *European Conference on Computer Vision*, 2010.
- A. Baumberg. Reliable feature matching across widely separated views. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 774–781, 2000.
- M. Bertalmío, G. Sapiro, V. Caselles, and C. Ballester. Image inpainting. In *Proceedings of SIGGRAPH*, pages 417–424, 2000.
- K. Bredies, K. Kunisch, and T. Pock. Total generalized variation. *SIAM Journal on Imaging Sciences*, 3(3):492–526, 2010.
- T. Brox, R. Boomgaard, F. Lauze, J. Weijer, J. Weickert, P. Mrázek, and P. Kornprobst. Adaptive structure tensors and their applications. *Visualization and Processing of Tensor Fields*, pages 17–47, 2006a.
- T. Brox, J. Weickert, B. Burgeth, and P. Mrázek. Nonlinear structure tensors. *Image and Vision Computing*, 24(1):41–55, 2006b.
- T. Brox, O. Kleinschmidt, and D. Cremers. Efficient nonlocal means for denoising of textural patterns. *IEEE Transactions on Image Processing*, 17(7):1083–1092, July 2008.
- A. Buades, B. Coll, and J.-M. Morel. A non local algorithm for image denoising. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 60–65, 2005. doi: 10.1109/CVPR.2005.38. <http://dx.doi.org/10.1109/CVPR.2005.38>.
- Antoni Buades, Bartomeu Coll, and Jean-Michel Morel. Non-Local Means Denoising. *Image Processing On Line*, 1, 2011. doi: 10.5201/ipol.2011.bcm_nlm.
- F. Calderero and V. Caselles. Multiscale analysis of images on riemannian manifolds. *SIAM Journal on Imaging Sciences*, 7(2):1108–1170, 2014.
- F. Cao, Y. Gousseau, S. Masnou, and P. Prez. Geometrically guided exemplar-based inpainting. *SIAM Journal on Imaging Sciences*, 4(4):

- 1143–1179, 2011.
- F. Catté, P.L. Lions, J.M. Morel, and T. Coll. Image selective smoothing and edge detection by nonlinear diffusion. *SIAM Journal on Numerical Analysis*, 29(1):182–193, 1992.
- A. Chambolle. An algorithm for total variation minimization and applications. *J. Math. Imaging Vis.*, 20(1-2):89–97, 2004.
- T. Chan and J. H. Shen. Mathematical models for local nontexture inpaintings. *SIAM J. App. Math.*, 62(3):1019–43, 2001.
- J. K. Cohen, F. G. Hagin, and J. B. Keller. Short time asymptotic expansions of solutions of parabolic equations. *Journal of Mathematical Analysis and Applications*, 38(1):82–91, 1972.
- A. Criminisi, P. Pérez, and K. Toyama. Region filling and object removal by exemplar-based inpainting. *IEEE Trans. on IP*, 13(9):1200–1212, 2004.
- K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. Image denoising by sparse 3-d transform-domain collaborative filtering. *Image Processing, IEEE Transactions on*, 16(8):2080–2095, Aug 2007. ISSN 1057-7149. doi: 10.1109/TIP.2007.901238.
- C. A. Deledalle, V. Duval, and J. Salmon. Non-local methods with shape-adaptive patches (nlm-sap). *Journal of Mathematical Imaging and Vision*, 43(2):103–120, 2012.
- L. Demanet, B. Song, and T. Chan. Image inpainting by correspondence maps: a deterministic approach. *Applied and Computational Mathematics*, 1100:217–50, 2003.
- D. L. Donoho and J. M. Johnstone. Ideal spatial adaptation by wavelet shrinkage. *Biometrika*, 81(3):425–455, 1994.
- I. Drori, D. Cohen-Or, and H. Yeshurun. Fragment-based image completion. In *ACM SIGGRAPH 2003 Papers*, volume 22, pages 303–12, 2003.
- A. A. Efros and T. K. Leung. Texture synthesis by non-parametric sampling. In *Proceedings of the IEEE ICCV*, pages 1033–38, September 1999.
- M. Elad and M. Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *Image Processing, IEEE Transactions on*, 15(12):3736–3745, Dec 2006. ISSN 1057-7149. doi: 10.1109/TIP.2006.881969.
- V. Fedorov and C. Ballester. Affine non-local means image denoising. *IEEE Transactions On Image Processing*, 2016. Submitted.
- V. Fedorov, P. Arias, R. Sadek, G. Facciolo, and C. Ballester. Linear multiscale analysis of similarities between images on riemannian manifolds:

- Practical formula and affine covariant metrics. *SIAM Journal on Imaging Sciences*, 8(3):2021–2069, 2015a. doi: 10.1137/141000002.
- V. Fedorov, G. Facciolo, and P. Arias. Variational framework for non-local inpainting. *Image Processing On Line*, 5:362–386, 2015b. doi: 10.5201/ipol.2015.136.
- V. Fedorov, P. Arias, G. Facciolo, and C. Ballester. Affine invariant self-similarity for exemplar-based inpainting. In *Proceedings of the 11th Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*, pages 48–58, 2016. doi: 10.5220/0005728100480058.
- A. Foi and G. Boracchi. Foveated self-similarity in nonlocal image filtering. *Human Vision and Electronic Imaging XVII*, 8291(1):829110, February 2012.
- A. Foi, V. Katkovnik, and K. Egiazarian. Pointwise shape-adaptive dct for high-quality denoising and deblocking of grayscale and color images. *IEEE Transactions on Image Processing*, 16(5):1395–1411, May 2007.
- H. Foroosh, J. B. Zerubia, and M. Berthod. Extension of phase correlation to subpixel registration. *IEEE Transactions on Image Processing*, 11(3):188–200, Mar 2002. ISSN 1057-7149. doi: 10.1109/83.988953.
- J. Garding and T. Lindeberg. Direct estimation of local surface shape in a fixating binocular vision system. In Jan-Olof Eklundh, editor, *Computer Vision ECCV '94*, volume 800 of *Lecture Notes in Computer Science*, pages 365–376. Springer Berlin Heidelberg, 1994. ISBN 978-3-540-57956-4. doi: 10.1007/3-540-57956-7_40. URL http://dx.doi.org/10.1007/3-540-57956-7_40.
- G. Gilboa and S. J. Osher. Nonlocal operators with applications to image processing. *Multiscale Modeling and Simulation*, 7(3):1005–1028, 2008.
- J. Gårding. Shape from texture for smooth curved surfaces in perspective projection. *Journal of Mathematical Imaging and Vision*, 2(4):327–350, 1992.
- S. Grewnig, S. Zimmer, and J. Weickert. Rotationally invariant similarity measures for nonlocal image denoising. *Journal of Visual Communication and Image Representation*, 22(2):117 – 130, 2011. ISSN 1047-3203. doi: <http://dx.doi.org/10.1016/j.jvcir.2010.11.001>. URL <http://www.sciencedirect.com/science/article/pii/S1047320310001458>.
- F. Guichard and J.M. Morel. Image analysis and PDEs. *book in preparation*, 2001.
- J. Hays and A.A. Efros. Scene completion using millions of photographs.

- In *SIGGRAPH*, New York, NY, USA, 2007. ACM.
- J.-B. Huang, J. Kopf, N. Ahuja, and S. B. Kang. Transformation guided image completion. In *International Conference on Computational Photography*, April 2013.
- J. B. Huang, S. B. Kang, N. Ahuja, and J. Kopf. Image completion using planar structure guidance. *ACM Transactions on Graphics (Proceedings of SIGGRAPH 2014)*, 33(4):129:1–129:10, 2014. doi: 10.1145/2601097.2601205. URL <http://doi.acm.org/10.1145/2601097.2601205>.
- R. Hummel. The scale-space formulation of pyramid data structures. In *Parallel computer vision*, pages 107–123. Academic Press Professional, Inc., 1987.
- R.A. Hummel, B. Kimia, and S.W. Zucker. Gaussian blur and the heat equation: forward and inverse solutions. In *IEEE International Conference on Computer Vision and Pattern Recognition*, pages 668–671, 1985.
- Kenichi Kanatani. *Group-theoretical methods in image understanding*, volume 2. springer-Verlag Berlin, 1990.
- N. Kawai, T. Sato, and N. Yokoya. Image inpainting considering brightness change and spatial locality of textures and its evaluation. In *Advances in Image and Video Technology*, pages 271–282. 2009.
- A. Kheradmand and P. Milanfar. A general framework for regularized, similarity-based image restoration. *IEEE Trans. on Image Processing*, 23(12):5136–5151, 2014.
- R. Kimmel, R. Malladi, and N. Sochen. Images as embedded maps and minimal surfaces: movies, color, texture, and volumetric medical images. *International Journal of Computer Vision*, 39(2):111–129, 2000.
- J.J. Koenderink. The structure of images. *Biological cybernetics*, 50(5):363–370, 1984.
- J.J. Koenderink and AJ Van Doorn. Dynamic shape. *Biological cybernetics*, 53(6):383–396, 1986.
- M. Lebrun, M. Colom, A. Buades, and J. M. Morel. Secrets of image denoising cuisine. *Acta Numerica*, 21:475–576, 5 2012. ISSN 1474-0508. doi: 10.1017/S0962492912000062. URL http://journals.cambridge.org/article_S0962492912000062.
- M. Lebrun, A. Buades, and J.-M. Morel. Implementation of the "Non-Local Bayes" (NL-Bayes) Image Denoising Algorithm. *Image Processing On Line*, 3:1–42, 2013a. doi: 10.5201/ipol.2013.16.
- M. Lebrun, A. Buades, and J. M. Morel. A nonlocal bayesian image de-

- noising algorithm. *SIAM Journal on Imaging Sciences*, 6(3):1665–1688, 2013b. doi: 10.1137/120874989. URL <http://dx.doi.org/10.1137/120874989>.
- M. Lebrun, M. Colom, and J.-M. Morel. Multiscale image blind denoising. *Image Processing, IEEE Transactions on*, 24(10):3149–3161, Oct 2015. ISSN 1057-7149. doi: 10.1109/TIP.2015.2439041.
- JP Lewis. Fast normalized cross-correlation. *Vision interface*, 10(1):120–123, 1995.
- T. Lindeberg. *Scale-space theory in computer vision*. Springer, 1993.
- D. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.
- J. Mairal, M. Elad, and G. Sapiro. Sparse representation for color image restoration. *Image Processing, IEEE Transactions on*, 17(1):53–69, Jan 2008. ISSN 1057-7149. doi: 10.1109/TIP.2007.911828.
- A. Mansfield, M. Prasad, C. Rother, T. Sharp, P. Kohli, and L. van Gool. Transforming image completion. In *Proceedings of BMVC*, pages 121.1–121.11, 2011.
- D. Marr and E. Hildreth. Theory of edge detection. *Proceedings of the Royal Society of London. Series B. Biological Sciences*, 207(1167):187–217, 1980.
- A. Martn, J. F. Garamendi, and E. Schiavi. Iterated rician denoising. In *In Proc. Int. Conf. Image Process. Comput. Vis. Pattern Recognit. IPCV-2011*, volume 11, pages 959–963, Las Vegas, USA, 2011.
- S. Masnou. Disocclusion: a variational approach using level lines. *IEEE Transactions on Image Processing*, 11(2):68–76, 2002.
- S. Masnou and J.-M. Morel. Level lines based disocclusion. In *Proceedings of IEEE ICIP*, volume 3, pages 259–263, 1998.
- J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide-baseline stereo from maximally stable extremal regions. *Image and Vision Computing*, 22(10):761–767, 2004.
- K. Mikolajczyk. Affine covariant features, dataset. <http://www.robots.ox.ac.uk/~vgg/research/affine/>, 2007.
- K. Mikolajczyk and C. Schmid. Scale & affine invariant interest point detectors. *International Journal of Computer Vision*, 60(1):63–86, 2004.
- J.M. Morel and G. Yu. Asift: A new framework for fully affine invariant image comparison. *SIAM Journal on Imaging Sciences*, 2(2):438–469, April 2009. ISSN 1936-4954.

- J.M. Morel and G. Yu. Is sift scale invariant? *Inverse Problems and Imaging*, 5(1):115–136, 2011.
- E. A. Nadaraya. On estimating regression. *Theory of Probability & Its Applications*, 9(1):141–142, 1964. doi: 10.1137/1109020. URL <http://dx.doi.org/10.1137/1109020>.
- M Nikolova. A variational approach to remove outliers and impulse noise. *Journal of Mathematical Imaging and Vision*, 20(1-2):99–120, 2004.
- P.J. Olver, G. Sapiro, and A. Tannenbaum. Differential invariant signatures and flows in computer vision: a symmetry group approach. Technical report, DTIC Document, 1993.
- D. Pavić, V. Schnefeld, and L. Kobbelt. Interactive image completion with perspective correction. *The Visual Computer*, 22(9-11):671–681, 2006. ISSN 0178-2789. doi: 10.1007/s00371-006-0050-2. URL <http://dx.doi.org/10.1007/s00371-006-0050-2>.
- P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 12(7):629–639, 1990a.
- P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 12(7):629–639, Jul 1990b. ISSN 0162-8828. doi: 10.1109/34.56205.
- G. Peyré. Manifold models for signals and images. *Computer Vision and Image Understanding*, 113(2):249–260, 2009.
- G. Peyré, S. Bougleux, and L. D. Cohen. Non-local regularization of inverse problems. In *Proceedings of the IEEE Conference on Computer Vision ECCV*, pages 57–68, 2008.
- N. Pierazzo, Rais Lebrun, M., M. E., J. M. Morel, and G. Facciolo. Non-local dual image denoising. In *In Image Processing (ICIP), 2014 IEEE International Conference on (pp. 813-817). IEEE.*, 2014.
- L. Pizarro, P. Mrázek, S. Didas, S. Grewenig, and J. Weickert. Generalised nonlocal image smoothing. *International Journal of Computer Vision*, 90:62–87, 2010.
- J. Portilla, V. Strela, M.J. Wainwright, and E.P. Simoncelli. Image denoising using scale mixtures of gaussians in the wavelet domain. *Image Processing, IEEE Transactions on*, 12(11):1338–1351, Nov 2003. ISSN 1057-7149. doi: 10.1109/TIP.2003.818640.
- M. Protter, M. Elad, H. Takeda, and P. Milanfar. Generalizing the non-local-means to super-resolution reconstruction. *IEEE Trans. on IP*, 18(1):36–51, 2009.

- L.I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(14):259 – 268, 1992. ISSN 0167-2789. doi: [http://dx.doi.org/10.1016/0167-2789\(92\)90242-F](http://dx.doi.org/10.1016/0167-2789(92)90242-F). URL <http://www.sciencedirect.com/science/article/pii/016727899290242F>.
- G. Sapiro and A. Tannenbaum. Affine invariant scale-space. *International Journal of Computer Vision*, 11(1):25–44, 1993.
- G. Sapiro and A. Tannenbaum. On affine plane curve evolution. *Journal of Functional Analysis*, 119(1):79–120, 1994.
- N. Sochen, R. Kimmel, and A.M. Bruckstein. Diffusions and confusions in signal and image processing. *Journal of Mathematical Imaging and Vision*, 14(3):195–209, 2001.
- J. L. Starck, E. J. Candès, and D. L. Donoho. The curvelet transform for image denoising. *IEEE Trans. on Image Processing*, 11(6):670–684, 2002.
- C. M. Stein. Estimation of the mean of a multivariate normal distribution. *The annals of Statistics*, 43(2):1135–1151, 1981.
- C. Tomasi and R. Manduchi. Bilateral filtering for gray and color images. In *Computer Vision, 1998. Sixth International Conference on*, pages 839–846. IEEE, 1998.
- T. Tuytelaars and K. Mikolajczyk. Local invariant feature detectors: A survey. *Foundations and Trends in Computer Graphics and Vision*, 3(3): 177–280, July 2008. ISSN 1572-2740.
- T. Tuytelaars and L. Van Gool. Matching widely separated views based on affine invariant regions. *International Journal of Computer Vision*, 59(1):61–85, August 2004. ISSN 0920-5691.
- S. R. S. Varadhan. On the behavior of the fundamental solution of the heat equation with variable coefficients. *Communications on Pure and Applied Mathematics*, 20(2):431–455, 1967.
- A. Vedaldi and B. Fulkerson. VLFeat: An open and portable library of computer vision algorithms. <http://www.vlfeat.org/>, 2008.
- P. Viola and W. M. Wells. Alignment by maximization of mutual information. In *Computer Vision, 1995. Proceedings., Fifth International Conference on*, pages 16–23, Jun 1995. doi: 10.1109/ICCV.1995.466930.
- L. Wang, Y. Zhang, and J. Feng. On the euclidean distance of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(8): 1334–1339, Aug 2005. ISSN 0162-8828. doi: 10.1109/TPAMI.2005.165.
- Z. Wang. Image affine inpainting. In *Image Analysis and Recognition*,

- volume 5112 of *Lecture Notes in Computer Science*, pages 1061–1070. 2008. ISBN 978-3-540-69811-1. doi: 10.1007/978-3-540-69812-8_106. URL http://dx.doi.org/10.1007/978-3-540-69812-8_106.
- Zhou Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, April 2004. ISSN 1057-7149. doi: 10.1109/TIP.2003.819861.
- Geoffrey S. Watson. Smooth regression analysis. *Sankhya: The Indian Journal of Statistics, Series A (1961-2002)*, 26(4):359–372, 1964. ISSN 0581572X. URL <http://www.jstor.org/stable/25049340>.
- J. Weickert. *Anisotropic diffusion in image processing*, volume 1. Teubner Stuttgart, 1998.
- J. Weickert. Coherence-enhancing diffusion filtering. *International Journal of Computer Vision*, 31(2):111–127, 1999.
- J. Weickert, S. Ishikawa, and A. Imiya. Linear scale-space has first been proposed in japan. *Journal of Mathematical Imaging and Vision*, 10(3): 237–252, 1999.
- Y. Wexler, E. Shechtman, and M. Irani. Space-time completion of video. *IEEE Transactions on PAMI*, 29(3):463–476, 2007.
- A. Witkin. Scale-space filtering: A new approach to multi-scale description. In *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP'84.*, volume 9, pages 150–153. IEEE, 1984.
- L.P. Yaroslavsky. Local adaptive image restoration and enhancement with the use of dft and dct in a running window, 1996. URL <http://dx.doi.org/10.1117/12.255218>.
- K.J. Yoon and I.S. Kweon. Adaptive support-weight approach for correspondence search. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28(4):650–656, April 2006. ISSN 0162-8828. doi: 10.1109/TPAMI.2006.70.
- G. Yu and G. Sapiro. Dct image denoising: a simple and effective image denoising algorithm. *Image Processing On Line*, 1, 2011. doi: <http://dx.doi.org/10.5201/ipol.2011.ys-dct>.
- G. Yu, G. Sapiro, and S. Mallat. Image modeling and enhancement via structured sparse model selection. In *In Image Processing (ICIP), 2010 17th IEEE International Conference on (pp. 1641-1644). IEEE.*, 2010.
- L. Zhang, W. Dong, D. Zhang, and G. Shi. Two-stage image denoising by principal component analysis with local pixel grouping. *Pattern Recognition*, 43(4):1531 – 1549, 2010. ISSN 0031-3203. doi: <http://dx.doi.org/>

10.1016/j.patcog.2009.09.023. URL <http://www.sciencedirect.com/science/article/pii/S0031320309003677>.

V.A. Zimmer and G. Piella. *An Adaptive Multiscale Similarity Measure for Non-rigid Registration*, pages 203–212. Springer International Publishing, Cham, 2014. ISBN 978-3-319-08554-8. doi: 10.1007/978-3-319-08554-8_21. URL http://dx.doi.org/10.1007/978-3-319-08554-8_21.

C. Zuo, L. Jovanov, H.Q. Luong, B. Goossens, W. Philips, Yu Liu, and M. Zhang. Rotation invariant similarity measure for non-local self-similarity based image denoising. In *Image Processing (ICIP), 2015 IEEE International Conference on*, pages 1618–1622, Sept 2015. doi: 10.1109/ICIP.2015.7351074.

By Vadim Fedorov and licensed under

Creative Commons Attribution-NonCommercial-NoDerivs 3.0 Unported



You are free to Share – to copy, distribute and transmit the work Under the following conditions:

- **Attribution** – You must attribute the work in the manner specified by the author or licensor (but not in any way that suggests that they endorse you or your use of the work).
- **Noncommercial** – You may not use this work for commercial purposes.
- **No Derivative Works** – You may not alter, transform, or build upon this work.

With the understanding that:

Waiver – Any of the above conditions can be waived if you get permission from the copyright holder.

Public Domain – Where the work or any of its elements is in the public domain under applicable law, that status is in no way affected by the license.

Other Rights – In no way are any of the following rights affected by the license:

- Your fair dealing or fair use rights, or other applicable copyright exceptions and limitations;
- The author's moral rights;
- Rights other persons may have either in the work itself or in how the work is used, such as publicity or privacy rights.

Notice – For any reuse or distribution, you must make clear to others the license terms of this work. The best way to do this is with a link to this web page.
