Visual Scene Analysis based on the Kurtosis of Responses of Independent Component Filters

ANDRE BORGES CAVALCANTE

Abstract

This thesis presents a new methodology for visual scene analysis. This methodology consists of analyzing the kurtosis of responses of independent component (IC) filters. Here, our methodology is applied on three problems or subjects of visual scene analysis. The first subject is *segmentation of depth-of-field images*. The second subject is *segmentation of nature-made and man-made structures*. The third subject is *measuring the perception of complexity in streetscapes*.

In the first studied subject of visual scene analysis, i.e., *segmentation of depth-of-field images*, the goal is to *recognize* and *segment* focused and unfocused regions in the visual scene. In our proposed method, two different sets of IC filters are united or joined together. One set of IC filters is designed to respond to focused image regions. The other set of IC filters is designed to respond to unfocused image regions. By measuring the kurtosis of responses of the whole filter population, our method is able to recognize when a region in the scene is focused or unfocused. In terms of objective criteria, our method exhibits the highest performance among the fast methods of segmentation of depth of field. The performance of our method is only lower in comparison to that of time-consuming methods.

In the second studied subject of visual scene analysis, i.e., *segmentation of natural and man-made structures*, the goal is to recognize and segment natural and artificial objects in the visual scene. Similar to the method of depth-of-field segmentation, two different sets of IC filters are united or joined together. One set of IC filters is designed to respond to natural objects. The other set of IC filters is designed to respond to artificial or man-made objects. By measuring the kurtosis of responses of the whole filter population, our method is able to recognize when between the two different types of objects. In comparison to other methods, our proposed system exhibits the highest performance in terms of objective criteria.

In the third studied subject of visual scene analysis, i.e., *measuring the perception of complexity in streetscapes*, the goal is to create a measure to quantify the human perception of visual complexity in streetscapes. Our proposed measure of complexity is based on the statistics of local contrast and spatial frequency of the visual scene. Notice that the kurtosis-based methodology proposed in this thesis is used to extract the statistics of local spatial frequency. In comparison to classic and new methods, the proposed measure of complexity exhibits higher correlation with the subjective opinion of human participants. Furthermore, we show that our measure can be used to analyze perception in nighttime images.

For all three subjects of visual scene analysis, this manuscript presents motivation, state of art, open problems, and finally our experiments and results using the proposed methodology. In this regard, our methods exhibit competitive or higher performance than that of the state of art for all subjects. It is also important to highlight that the proposed methodology is computationally simple and fast. Therefore, our methodology is attractive for industry and consumer real-time vision applications. Notice that we also clearly present the shortcomings of our methods for each studied application.

keywords: visual scene analysis, segmentation, independent component analysis, depth-of-field, man-made object, visual complexity, streetscapes

Contents

To my father, mother and sister.

Chapter 1

Introduction

1.1 Motivation and Objective

Visual scene analysis may be divided into two perceptual processes [1]. The first process is *segmentation*, i.e., segregating the scene into different partitions or regions. The second perceptual process is *recognition*, i.e., classifying the scene or the scene regions into categories defined a priori.

In scientific literature and in the industry, many computer methods have been proposed for visual scene analysis. The number of applications of such technologies is extensive. For instance, we can cite applications such as visual inspection, text and document handling, surveillance, medical image processing and assisted diagnostic, biometrics, object detection and tracking, etc.

Visual scene analysis can also be used for *environmental understanding*, i.e., understanding the physical features of the environment. In robotics, environmental understanding is required for applications related to autonomous navigation and driving.

Besides the large number of applications, another motivation is the large number of platforms implementing methods of visual scene analysis. For instance, visual scene analysis is implemented in platforms such as automotive, airborne and space, medical equipment, and general low-power consumer electronics such as mobile computers.

Those platforms involve different trade-offs between energy consumption, time consumption, memory consumption, size and weight of the processor device. In this regard, methodologies of visual scene analysis which are computationally complex may not conform to the consumption requirements.

Therefore, there is a need for studies which focus on developing computationally low-cost methods of visual scene analysis. The goal of this thesis is exactly to introduce a original and simple methodology which can be used for both segmentation and recognition. The proposed methodology consists of extracting and analyzing the kurtosis of responses of independent component filters. Figure 1.1 illustrates the basic idea of the proposed methodology. The details of this methodology are fully described in the next chapter.

Based on the proposed methodology, we have built new computer methods for subjects or applications of visual scene analysis. These subjects are presented later in this chapter. For reproducibility or extension of this research, data and software source-code are made available on-line at `http://github.com/andrecavalcante/thesis/`.

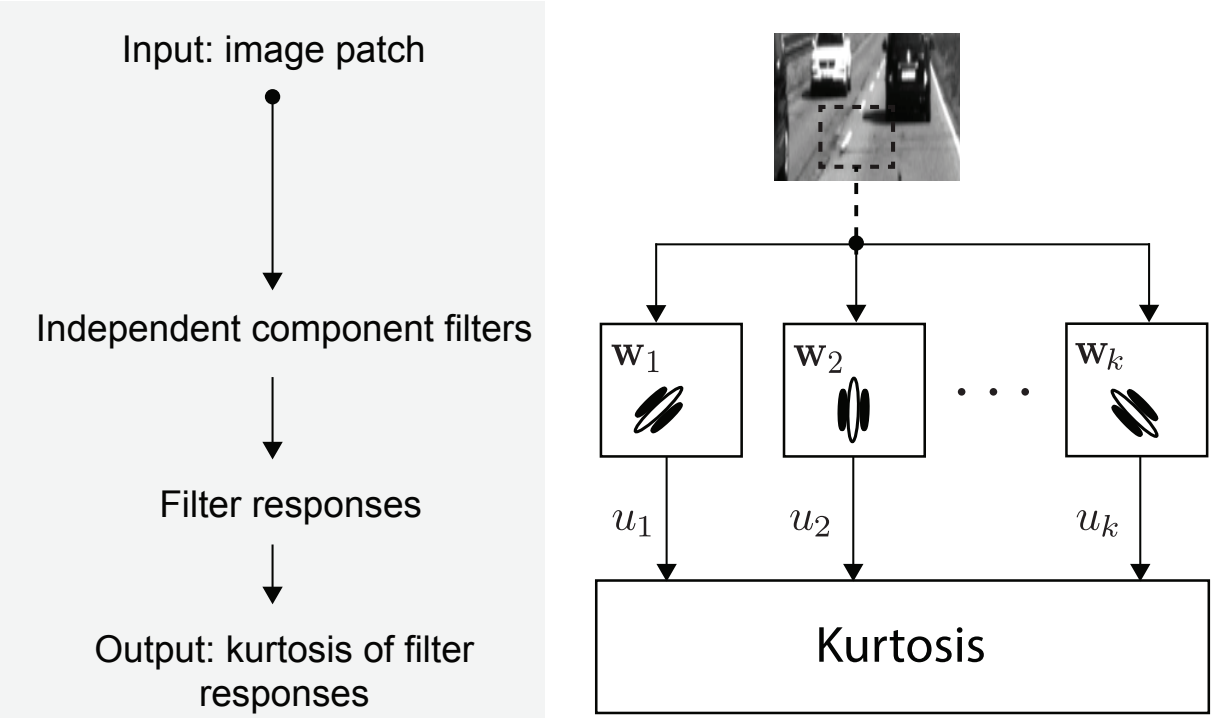

Figure 1.1: **Basic view of the proposed methodology**. Firstly, the responses of independent component filters are calculated for an input scene. Then, the kurtosis (i.e., the normalized fourth-order moment) of the filter responses is extracted and analyzed.

1.2 Related work

Image segmentation techniques are mainly classified as region-based methods or boundary detection methods [2]. In region-based methods, different image regions are defined based on the homogeneity or similarity between neighboring pixels. This similarity is computed as a function of image characteristics such as pixel intensity, color, etc. Notice that determining the best function to cluster pixels or form regions may be considered as pattern recognition problem. In this regard, cluster algorithms such as k-means, expectation maximization and mean shift clustering may be considered region-based methods when applied for image segmentation. In fact, one should notice that the methodology proposed in Figure 1.1 is a region-based methodology.

Another important aspect of region-based methods is how the segmentation process is executed in space. In this regard, region-based methods may be classified as region-growing methods or region-split methods [3, 4]. In region-growing methods, a pixel is chosen as starting point of a region. Then, additional neighboring pixels are added based on the similarity function. In split methods, the whole image is considered as one region. If all pixels within this region does not satisfy the similarity criteria, then the region is split into subregions. This process is then repeated.

Being different from region-based methods, boundary detection methods focus on detecting the boundaries between image regions. In order to detect boundaries, these methods may also use simple image characteristics such pixel intensity. One example is the watershed segmentation method [5]. In this technique, an image is interpreted as a topographic relief. Specifically, pixel intensities are considered as altitudes or elevations

in the relief. Boundaries are then detected by gradient or "flooding" techniques applied on the relief. Another example is the method quite recently proposed to recover occlusion boundaries from single image [6]. In this method, a very large number of 2D and 3D clues are used to characterize occlusion. Based on this clues, a conditional random field model is trained to determine the likelihood of boundaries in the image.

In case of image recognition, methods can be mainly classified as template matching methods, statistical approaches and neural networks based methods [7]. Template matching methods are based on predefined templates for each possible image category. These templates are images or feature vectors which represent the main characteristics of the categories to be recognized. Template methods work by measuring the similarity between the image and templates. The similarity is usually computed as the correlation or some distance metric between the image and the template.

Defining the template and the similarity function is an important issue in template matching. However, template match research strongly focus on increasing the speed of the matching procedure itself [8]. Examples are fast matching based on hierarchical partitioning [9] and based on projection kernels [10].

Statistical methods generally work by calculating the probabilities of an image or image pixels belonging to different categories [7]. The recognition procedure in these methods is usually divided into a feature extraction phase and a learning or classification phase. Many methods have been proposed for feature extraction. For instance we can cite principal component analysis, linear discriminant analysis, independent component analysis, self-organizing maps, etc. Many methods have also been proposed for learning or classification, for instance, logistic classifier, Fisher linear discriminant, support vector classifier, Bayes plug-in, conditional random field, radial basis network, etc.

Neural networks methods use a massive parallel processing architecture formed by artificial neurons. Such architecture is able to learn and generalize different patterns for later inference or recognition. Recently, works have been proposed for recognition, sometimes called deep-learning systems. For instance, a method for face recognition using 3D face modeling and a deep neural network which consists of nine layers involving more than 120 million parameters [11].

There are also methodologies of visual scene analysis which are similar to that proposed in Figure 1.1. One example is the methodology proposed by Haralick et al. [12]. Their methodology consists of extracting and analyzing the statistics of the distribution of intensity spatial co-occurrence. This methodology has been largely used for texture segmentation and recognition [13].

Another methodology widely used for visual scene analysis is the selective visual attention model proposed by Kock et al. [14, 15]. In this methodology, the input image is filtered in different frequency, color and orientation channels. Conspicuity maps (frequency map, color map and orientation map) are generated and transformed into a unique saliency map. Many computer methods have been proposed based on this methodology.

Another similar methodology is proposed by Hansen et al. [16]. This method consists of calculating the sum of thresholded responses of log-Gabor filters. They have used this method to characterize the structure of visual scenes. Besides these methodologies, we are going to cite other related works throughout this thesis.

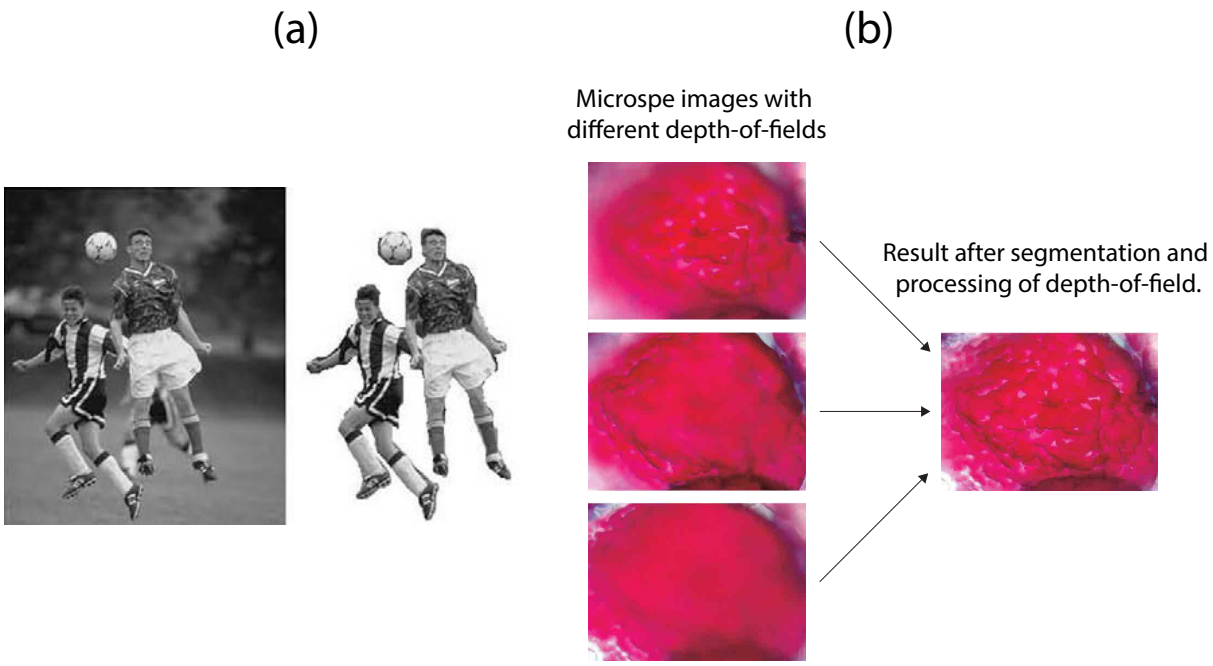

Figure 1.2: **Applications of depth-of-field image segmentation**. (a) Background removal [17]. (b) Extension or enhancement of depth-of-field in microscopy images [18].

1.3 Subjects studied in the thesis

This section presents three subjects or applications in which the proposed methodology (Figure 1.1) is used.

1.3.1 Segmentation of depth-of-field image

Depth-of-field (DOF) is a photographic technique generally used to separate an object of interest from background. This separation is achieved by focusing the camera sensor only on the object of interest. In the problem of segmentation of DOF images, the goal is to use image processing to extract the object of interest.

Segmentation of DOF image can be applied in amateur and professional photography. Specifically, DOF image segmentation can be used to remove or change the background of a scene. This process can be easily implemented in smartphones and other computer systems. Figure 1.2(a) illustrates this application.

Segmentation of DOF image is also commonly applied in microscopy image analysis [18, 19]. Specifically, DOF image segmentation is used to enhance biological specimen's profile in microscopy imaging. During a microscopy imaging test or experiment, depth-of-field may change for different profile images. Segmentation of depth-of-field is used to select areas of the profile that are in focus. Then, focused areas are combined to form a clear profile image. Figure 1.2(b) illustrates this application.

1.3.2 Segmentation of natural and man-made structures

Here, the goal is to classify visual objects into two categories: nature-made or man-made. One should notice the importance of segmentation of natural and man-made

structures for real-world artificial vision systems. Specifically, this type of segmentation can be used as a preprocessing step for many vision applications. For example, the vision system of an autonomous machine might execute tasks such as people detection, building detection, car detection, street sign detection, lane detection, etc. For these tasks, spending battery power and computer resources for processing image areas with purely vegetation or sky information seems unprofitable. Therefore, segmentation of natural and man-made structures can be used to select image areas of potential interest and save computer resources.

This type of segmentation has also been applied in several areas such as autonomous navigation [20–22], urban planning [23, 24], damage assessment [25–27], and military applications such as terrain and underwater surveillance [28, 29], mapping and reconnaissance [30, 31], and target detection [32, 33]. In this section, we briefly describe each case.

In autonomous navigation supported by aerial data, the segmentation of natural and man-made structures has been used for terrain classification [22]. In aerial images, the terrain contains roads and buildings which can be used for determining route and destination. However, such information may be found corrupted by bodies of vegetation and "off-road" areas. In this case, it is important to proper segment those structures.

Furthermore, this type of segmentation has been used for detecting vegetation obstacles which can be driven over by vehicles [20, 21]. Specifically, obstacles such as tall grass, large bushes and other types of vegetation are not rigid. In this case, the autonomous vehicle does not need to change route nor avoid the obstacle. Figure 1.3(a) shows illustrations of the above applications on autonomous navigation.

In urban planning, the segmentation of natural and man-made structures has been used for analyzing urban spaces [23, 24]. Specifically, this type of segmentation is used to extract information about road networks and buildings from Geographic Information Systems (GIS) image data [23]. Also, it has been used to segment rural regions from urban areas from single aerial images [24]. Furthermore, it can be used to evaluate changes in the amount of vegetation in rural and urban areas (e.g., deforestation).

The segmentation of natural and man-made structures has also been used in damage assessment after natural disasters [25–27]. Specifically, the segmentation is applied on pre-event images and post-event images to detect potential changes in natural and man-made structures present in the environment. Figure 1.3(b) and 1.3(c) shows illustrations of the above applications in urban planning and damage assessment, respectively.

In military applications, the segmentation of natural and man-made structures is the core of many systems. For instance, this type of segmentation has been used to detect patterns associated with civil construction in desert areas [28, 31]. Furthermore, it has been used to analyze underwater sonar images to detect man-made objects such as mines [29]. Finally, many target detection and tracking systems are based on segmenting man-made objects from natural backgrounds [32, 33]. Figure 1.3(d) shows illustrations of the applications in the military domain.

1.3.3 Measuring the perception of complexity in streetscapes

Here, the goal is to create a computer method to quantify the visual complexity perceived in streetscapes. Streetscapes are a specific category of urban scenes. According to Rapoport [34], a streetscape is a more or less narrow and straight urban space lined up by buildings, used for circulation and other activities. Figure 1.4(a) shows an illustration

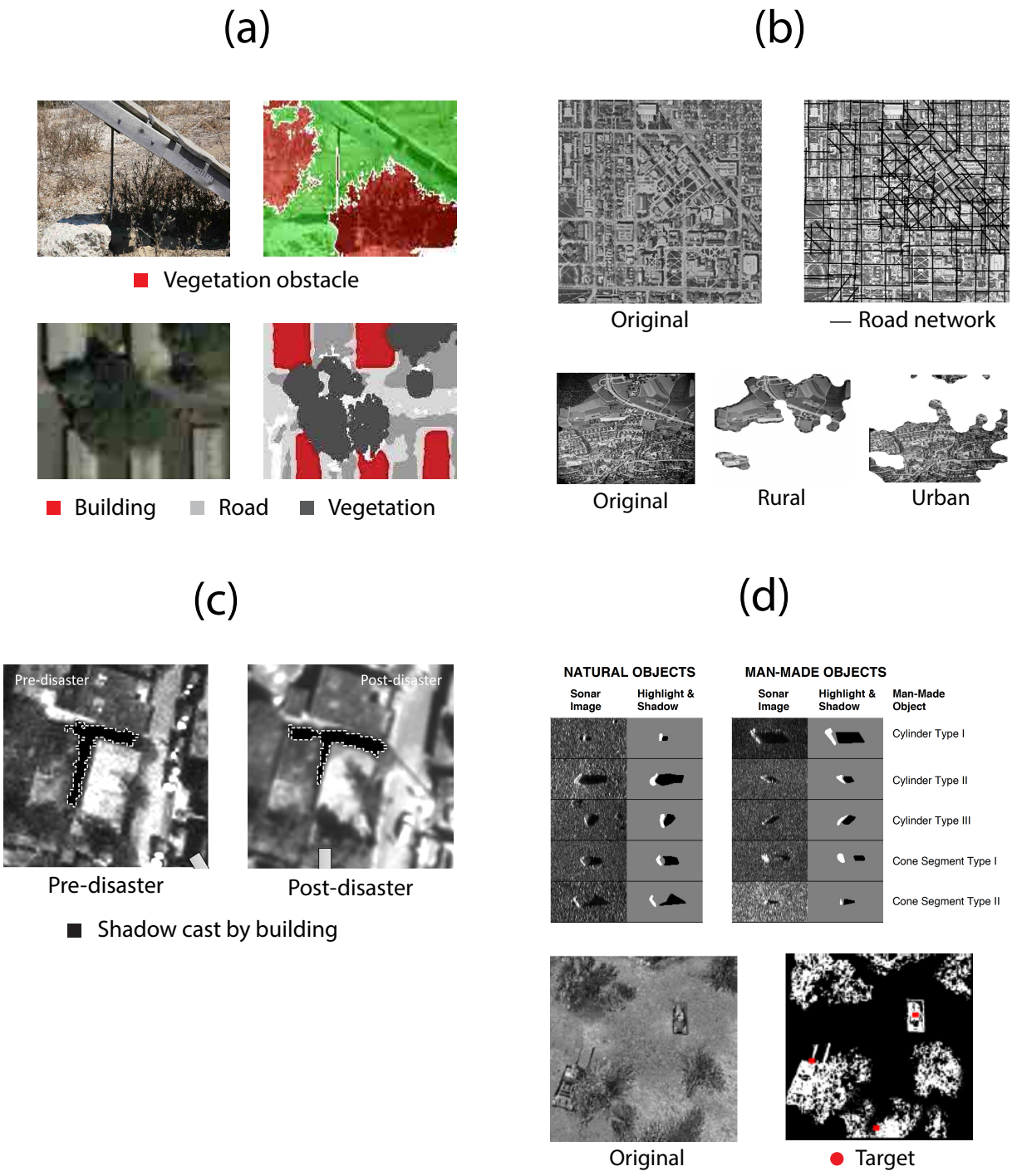

Figure 1.3: **Application of segmentation of natural and man-made structures**. Images were extracted from the following researches: (a) Autonomous navigation [20, 22]. (b) Urban planning [23, 24]. (c) Damage assement [25]. (d) Military applications [29, 32].

(a) (b)

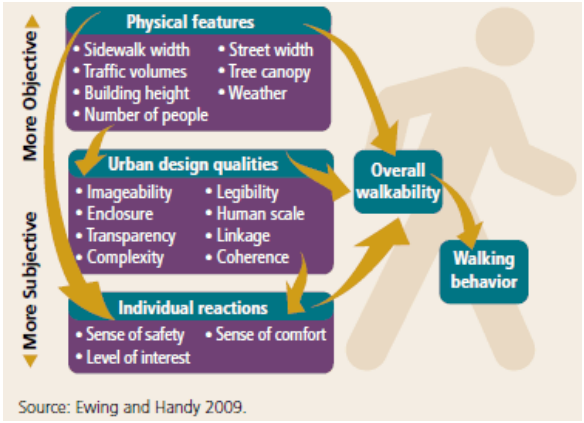

Figure 1.4: **The characteristics of streetscapes**. (a) Illustration of a streetscape (extracted from Google). According to Rapoport [34], a streetscape is a more or less narrow and straight urban space lined up by buildings, used for circulation and other activities. (b) Objective and subjective characteristics of streetscapes. Image extracted from [35].

of a streetscape.

Figure 1.4(b) shows examples of objective and subjective characteristics of streetscapes. It has been suggested that these characteristics directly influence how people locally behave in the city [35, 36]. Visual complexity is one example of subjective characteristic.

In general, the complexity of the environment heavily influences perception processes such as visual attention and visual search [37–41]. Specifically, visual complexity generally reflects the amount of time, eye fixations, and difficulty of finding an object in the environment [37–41].

These perceptual processes are fundamental for activities performed at streetscapes such as driving vehicles and shopping. Let's briefly explore how streetscape complexity influences these activities.

During driving, streetscape complexity affects the perception of driving speed, choice of driving speed, perception of safety and hazard, time for peripheral detection and time for reaction [43–48]. Specifically, driving speed is generally perceived higher in complex streetscapes. In this situation, drivers tend to decrease speed as a form of compensation. Furthermore, there is an additional issue related to driving in complex streetscapes. Namely, complex streetscapes make drivers slower to detect and react to events occurring at the peripheral vision [44].

Notice that the effects of visual complexity in driving are independent of on-street parking [45]. Specifically, it is suggested that drivers anticipate the potential activity of pedestrians in commercial environments, or the contribution of purely visual components of a complex environment (such as increased optical flow), or some combination of these effects [45].

For example, Figure 1.5 shows streetscapes with increasing complexity (in this case, images were evaluated only by the author). According to [45], the increasing complexity evokes different perceptions for hazard and potential activity of pedestrians at those

Figure 1.5: **Streetscapes with increasing complexity**. Complexity increases from the top to the bottom. Images extracted from the KITTI Vision Benchmark Suite [42].

streets. In case this hypothesis is true, a computer method created to measure the complexity of streetscapes could be used in driving applications. For instance, the method could allow robots and autonomous vehicles anticipate potential activity of pedestrians (similarly to the ability of human drivers).

In fact, the complexity of streetscapes is known to influence the interest and preference of pedestrians. Specifically, people express higher interest for streetscapes perceived higher in complexity [35, 36, 49, 50].

In the field of urban planning, researchers try to understand how to properly increase the complexity of streetscapes. Their goal is to increase interest and hold the attention of pedestrians. This is important for streetscapes in commercial districts. In this regard, a computer method which mimics our perception of complexity can be useful for automatic analysis of streetscapes.

1.4 Thesis characteristics

1.4.1 Originality and contributions

1. A new methodology for visual scene analysis. This new methodology is general and it can be applied in many subjects or problems of visual scene analysis.

2. A new computer method for *segmentation of depth-of-field image*.

3. A new computer method for *segmentation of natural and man-made structures*.

4. A new computer method for *measuring the perception of complexity in streetscapes*.

5. In regard of *measuring the perception of complexity in streetscapes*, our results reveal mechanisms related to the perception of visual complexity. Furthermore, our proposed measure of visual complexity is a helpful tool for research studies on visual perception.

6. For the fields of visual neuroscience and image coding, we provide an original analysis of differences between IC filters and Gabor functions. We show how these differences are related to the receptive fields of cells from the primary visual cortex. Also, we show how these differences influence error in image coding.

1.4.2 Chapter overview

This manuscript is organized according to the outline presented in Figure 1.6. Chapter 2 presents theoretical background of independent component filters and kurtosis. Furthermore, Chapter 2 provides the complete description of the proposed methodology for visual scene analysis, which is illustrated in Figure 1.1.

Chapter 3 presents the application of our methodology for *segmentation of depth-of-field images*. Firstly, Chapter 3 describes state-of-art and open problems of DOF segmentation. Secondly, Chapter 3 describes the details of our method for DOF segmentation. Finally, experiments, results and remarks are presented.

Chapter 4 presents the application of our methodology for *segmentation of natural and man-made structures in streetscapes*. For this subject, state-of-art and open problems are

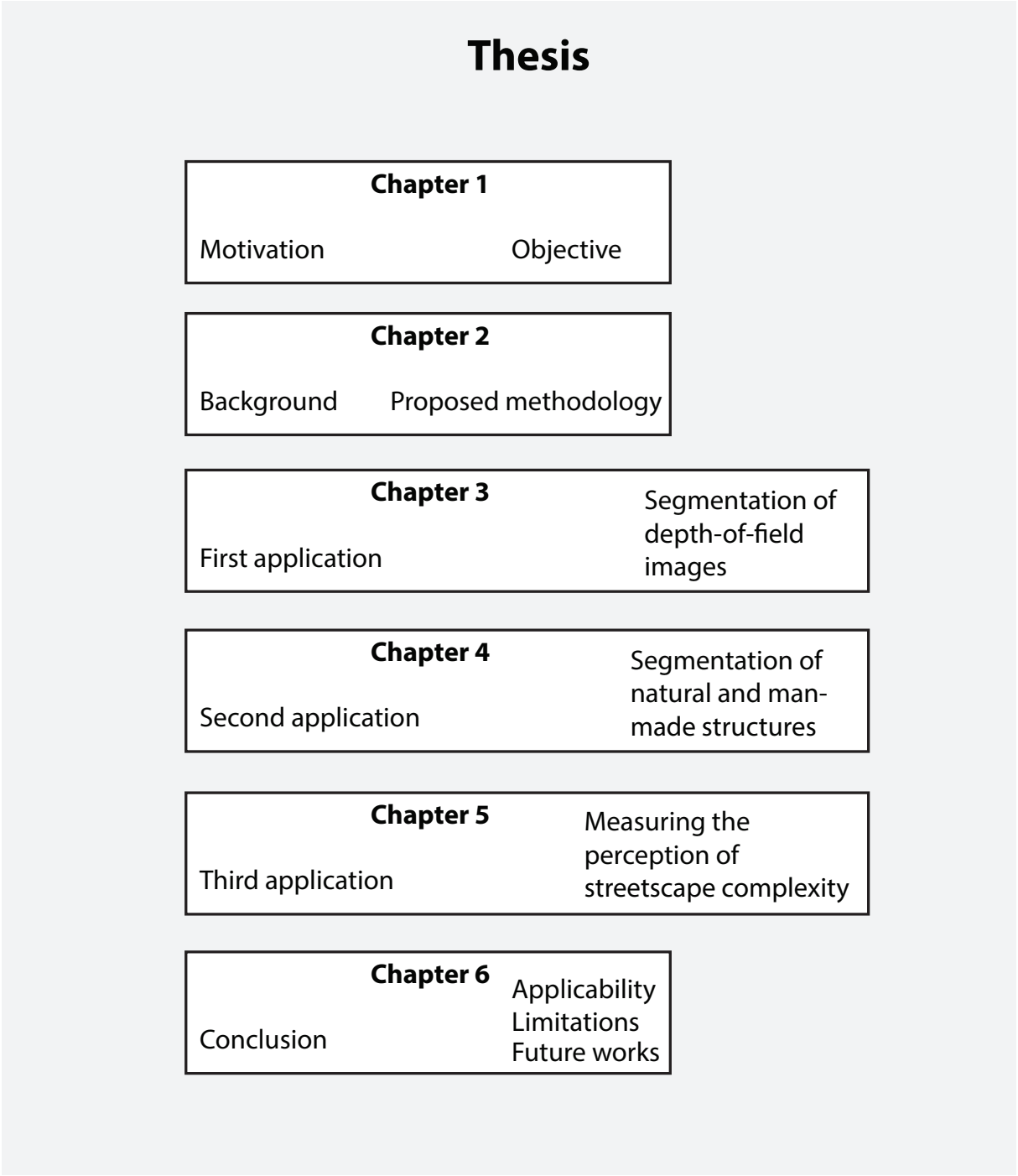

Figure 1.6: **Chapter overview**. The first chapter states the motivation and objective of this thesis on visual scene analysis. The second chapter introduces the necessary concepts and formally presents the proposed methodology of visual scene analysis. Chapters three, four and five reports three different applications of the proposed methodology. Chapter six summarises the thesis presenting applicability and limitation issues of the proposed methodology and future works.

also presented. Furthermore, the proposed method is described and experiments, results and remarks are presented.

Chapter 5 presents the application of our methodology for *measuring the perception of complexity in streetscapes*. Chapter 5 briefly introduces the research field of perception of visual complexity. Specifically, this chapter discusses works which pioneered computational modeling of perception of complexity. Furthermore, Chapter 5 discusses studies about perception of complexity in urban environments such as streetscapes. In this regard, this chapter states open problems of measuring the perception in streetscapes. Finally, Chapter 5 presents the proposed method, experiments, results and remarks.

Chapter 6 presents a summary and discussion about the use of the proposed methodology in visual scene analysis, especially regarding the subjects of Chapter 3, 4 and 5. Furthermore, Chapter 6 discusses shortcomings of the methodology, and also, shortcomings of the overall work carried out by us. Finally, this chapter presents the future works for this research.

Appendix sections are introduced in the end of this thesis. The appendixes contain mathematical proofs, derivation of algorithms and computer methods previously proposed by the author for visual scene analysis. These appendix sections are referenced as needed throughout the text.

Chapter 2

Proposed methodology

2.1 Background

2.1.1 Independent component filters

This section explains how independent component filters are learned. In *independent component analysis* (ICA), input variables $x_1, x_2, \ldots, x_n \in \mathbb{R}$ are transformed into output variables $y_1, y_2, \ldots, y_n \in \mathbb{R}$. The goal of ICA is to maximize the mutual statistical independence among variables $y_1, y_2, \ldots, y_n$. The ICA transformation is executed as follows.

The first step consists of transforming the input variables $x_1, x_2, \ldots, x_n$ into mutually uncorrelated variables $z_1, z_2, \ldots, z_n \in \mathbb{R}$. This can be represented as

$$z_i = \mathbf{v}_i^{\mathrm{T}} \mathbf{x}, \tag{2.1}$$

where $\mathbf{x} = [x_1 x_2 \ldots x_n]^{\mathrm{T}}$ and vectors $\mathbf{v}_i \in \mathbb{R}^n$ represent the *decorrelation* transformation. For the FastICA algorithm [51], vectors $\mathbf{v}_i$ are computed as

$$\mathbf{v}_i = d_i^{-\frac{1}{2}} \mathbf{e_i}, \tag{2.2}$$

where coefficients $d_1, d_2, \ldots, d_n \in \mathbb{R}$ and vectors $\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_n \in \mathbb{R}^n$ represent the eigenvalues and eigenvectors of the covariance matrix of the input variables, respectively.

The second step consists of transforming variables $z_1, z_2, \ldots, z_n$ into variables $y_1, y_2, \ldots, y_n \in \mathbb{R}$ according to

$$y_i = \mathbf{h}_i^{\mathrm{T}} \mathbf{z}, \tag{2.3}$$

where $\mathbf{z} = [z_1 z_2 \ldots z_n]^{\mathrm{T}}$ and $\mathbf{h}_i \in \mathbb{R}^n$. For the FastICA algorithm, vectors $\mathbf{h}_i$ are computed by the following iterative rule.

$$\mathbf{h}_i \leftarrow E\{\mathbf{z} g(\mathbf{h}_i^{\mathrm{T}} \mathbf{z})\} - E\{g\prime(\mathbf{h}_i^{\mathrm{T}} \mathbf{z})\} \mathbf{h}_i, \tag{2.4}$$

where $g(.) = tanh(.)$. The derivation of Equation (2.4) is demonstrated in Appendix A. After one iteration of (2.4), vectors $\mathbf{h}_i$ should be normalized and orthogonalized, i.e.,

$$\mathbf{h}_i \leftarrow \frac{\mathbf{h}_i}{\|\mathbf{h}_i\|}; \tag{2.5}$$

$$\mathbf{h}_i \leftarrow \mathbf{h}_i - \sum_{k=1}^{i-1} (\mathbf{h}_i^{\mathrm{T}} \mathbf{h}_k) \mathbf{h}_k. \tag{2.6}$$

After convergence of (2.4), vectors $\mathbf{h}_i$ can be used in Eq. (2.3) to obtain y_i. Only in this case, variables $y_1, y_2, \ldots, y_n$ become maximally mutually statistically independent. This completes the ICA procedure for the FastICA algorithm.

Now, denoting $\mathbf{h}_i = [h_{i1}h_{i2}\ldots h_{in}]^{\mathrm{T}}$, Eq. (2.3) can be rewritten as

$$y_i = h_{i1}z_1 + h_{i2}z_2 + \cdots + h_{in}z_n. \tag{2.7}$$

Substituting Eq. (2.1) into Eq. (2.7),

$$y_i = h_{i1}\mathbf{v}_1^{\mathrm{T}}\mathbf{x} + h_{i2}\mathbf{v}_2^{\mathrm{T}}\mathbf{x} + \cdots + h_{in}\mathbf{v}_n^{\mathrm{T}}\mathbf{x}. \tag{2.8}$$

Denoting $\mathbf{v}_i = [v_{i1}v_{i2}\ldots v_{in}]^{\mathrm{T}}$, Eq. (2.8) can be written as

$$\begin{aligned} y_i = h_{i1}[v_{11}x_1 + v_{12}x_2 + \cdots v_{1n}x_n] \\ +h_{i2}[v_{21}x_1 + v_{22}x_2 + \cdots v_{2n}x_n] \\ \vdots \\ +h_{in}[v_{n1}x_1 + v_{n2}x_2 + \cdots v_{nn}x_n]. \end{aligned} \tag{2.9}$$

Grouping x_i terms in Eq. (2.9),

$$\begin{aligned} y_i = x_1[h_{i1}v_{11} + h_{i2}v_{21} + \cdots h_{in}v_{n1}] \\ +x_2[h_{i1}v_{12} + h_{i2}v_{22} + \cdots h_{in}v_{n2}] \\ \vdots \\ +x_n[h_{i1}v_{1n} + h_{i2}v_{2n} + \cdots h_{in}v_{nn}]. \end{aligned} \tag{2.10}$$

Let us denote

$$w_{ij} = h_{i1}v_{1j} + h_{i2}v_{2j} + \cdots + h_{i2}v_{nj}. \tag{2.11}$$

Using Eq. (2.11) into Eq. (2.10),

$$y_i = w_{i1}x_1 + w_{i2}x_2 + \cdots + w_{in}x_n. \tag{2.12}$$

Finally, for $\mathbf{w}_i = [w_{i1}w_{i2}\ldots w_{in}]^{\mathrm{T}}$,

$$y_i = \mathbf{w}_i^{\mathrm{T}}\mathbf{x}. \tag{2.13}$$

Here, vectors $\mathbf{w}_i$ are called independent component (IC) filters. They represent the full ICA transformation from input variables $x_1, x_2, \ldots, x_n$ to output variables $y_1, y_2, \ldots, y_n$. Notice that from (2.11), IC filters could be written as

$$\mathbf{w}_i = h_{i1}\mathbf{v}_1 + h_{i2}\mathbf{v}_2 + \cdots + h_{in}\mathbf{v}_n. \tag{2.14}$$

2.1.2 Kurtosis

Kurtosis is generally defined as the normalised fourth-order moment of a random variable. For instance, let's assume that $u_1, u_2, \ldots, u_k \in \mathbb{R}$ represent k realizations of a random variable U. The *sample* kurtosis for a random variable U is represented here by K_U, and computed as

$$K_U = \frac{\frac{1}{k}\sum_{i=1}^{k}[u_i - \mu_U]^4}{\left\{\frac{1}{k}\sum_{i=1}^{k}[u_i - \mu_U]^2\right\}^2}, \tag{2.15}$$

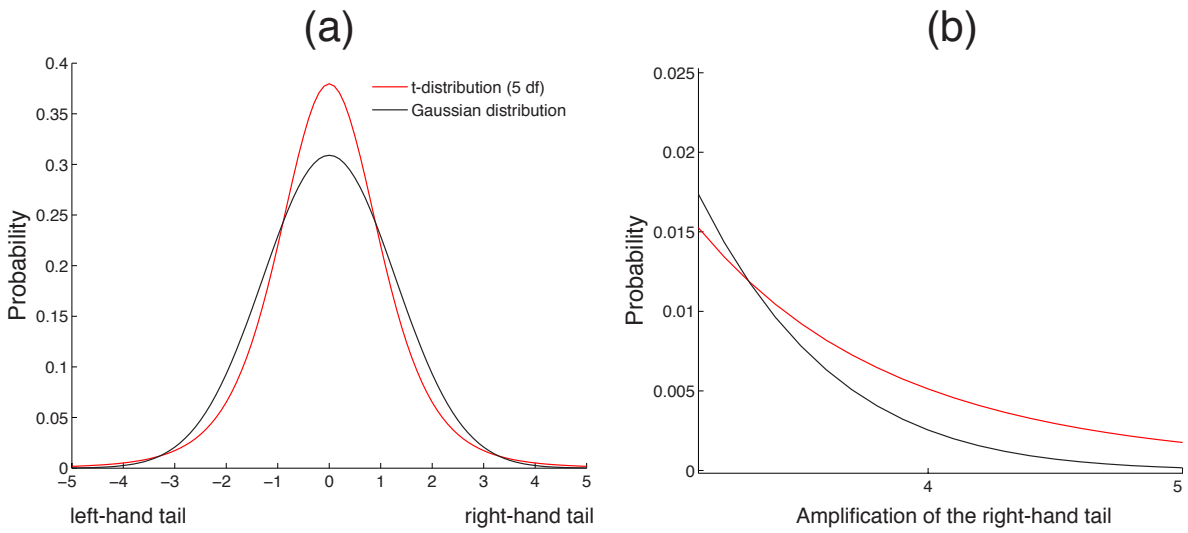

Figure 2.1: **Analysis of center and tails of well-known probability distributions**. (a) Plots of Gaussian and Student's-t probability distributions. The distributions have the same mean and variance. The t-distribution has five degrees of freedom (5 df). (b) Magnification of the right-hand tails. In comparison to the Gaussian distribution, the t-distribution has a higher peak and thicker tails.

where μ_U is the sample mean of U, i.e.,

$$\mu_U = \frac{1}{k}\sum_{i=1}^{k} u_i. \tag{2.16}$$

Dyson and Finucan showed that the higher the peak and tails of a distribution, the higher kurtosis [52,53]. For instance, let's analyze peak, tails and kurtosis for well-known probability distributions, i.e., the Gaussian distribution and the t-distribution.

These two distributions are shown in Figure 2.1. Notice that both distributions have the same mean and variance. In comparison to the Gaussian distribution, the t-distribution has a higher center peak and higher tails.

In accordance with higher peak and thicker tails, kurtosis of Student's distribution is higher than that of Gaussian distribution. Specifically, the kurtosis of these distributions can be calculated analytically. Appendix B of this thesis shows the calculation of kurtosis for Gaussian and Student's-t distributions. From Appendix B, kurtosis of Gaussian distribution is 3. Kurtosis of t-distribution with five degrees of freedom is 9.

The consequences of lower or higher kurtosis (or lower or higher peak and tails) can be observed on the following experiment. Figure 2.2 shows random numbers generated from Gaussian and t-distributions in Figure 2.1. From each distribution, 10,000 numbers are generated.

For both distributions, the majority of generated numbers have amplitude in the range $[-5, 5]$. However, t-distribution generated numbers of very high amplitude. The Gaussian distribution does not generate such values because probability at the tail is very low (See Figure 2.1(b)).

Thicker tails might suggest that the variance of t-distribution is higher than that of Gaussian distribution. In this case, however, variances are equal because t-distribution

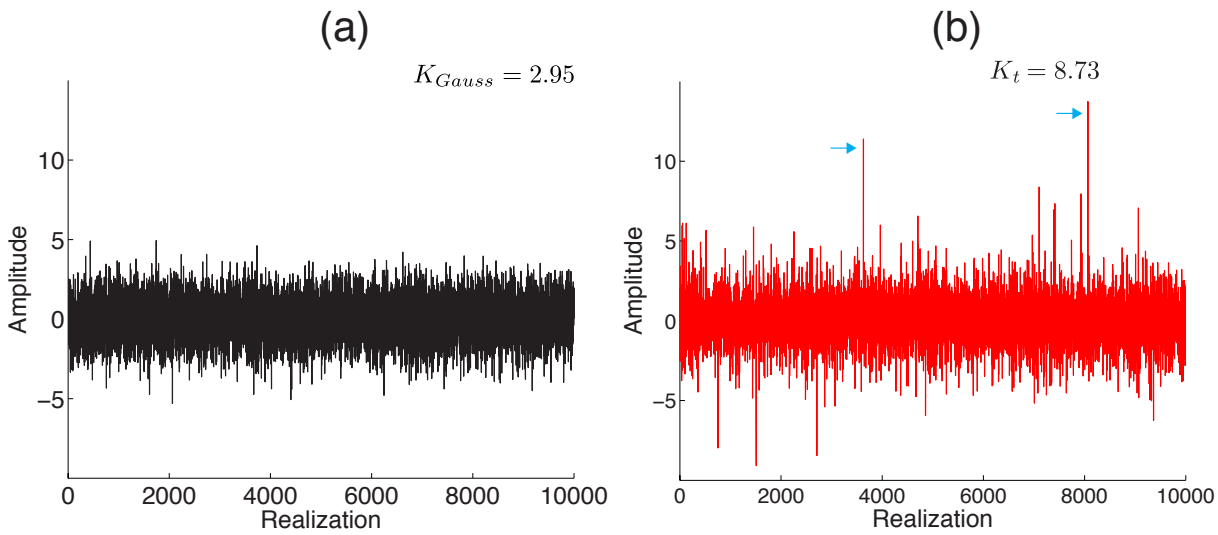

Figure 2.2: **Random numbers generated from the probability distributions**. (a) 10.000 realizations from Gaussian distribution. (b) 10.000 realizations from Student's-t distribution (5 df). Both probability distributions have equal mean and variance. For both distributions, the majority of numbers have amplitude in the range $[-5, 5]$. However, t-distribution generated numbers of very high amplitude. Examples of these very high amplitude realizations are indicated by blue arrows on plot (b). K_{Gauss} and K_t are the sample kurtosis of each set of realizations. Here, Matlab is used for random number generation.

has a center peak higher than that of Gaussian distribution. If only tails were higher (i.e., without higher peak), then t-distribution would have higher variance [54, 55].

In Figure 2.2, the sample kurtosis of random numbers generated from Gaussian distribution is $K_{Gauss} = 2.95$. Meanwhile, the sample kurtosis of random numbers generated from Student's-t distribution is $K_t = 8.73$. These kurtosis values are close to those calculated analytically.

Notice that since variance is the same, the kurtosis value must be used to discriminate or distinguish the pattern in Figure 2.2(b) from that in Figure 2.2(a).

2.2 The proposed method

This section formalizes the proposed methodology. The goal of our methodology is to determine if an image exhibits either a visual characteristic "A" or a visual characteristic "B". Here, "A" and "B" could represent a simple image characteristic such as spatial orientation. For example, if A = 90 degrees and B = 30 degrees, then our goal is to determine if the image exhibits a dominant spatial orientation of 90 degrees or 30 degrees.

Instead of spatial orientation, characteristics "A" and "B" could represent spatial frequency. For example, if A = high-frequency and B = low-frequency, our goal is to determine if the image exhibits either a high-frequency-like visual structure or a low-frequency-like visual structure. On the other hand, "A" and "B" could represent more complex characteristics such as image category. For example, if A = nature-made and B = man-made, then our goal is to determine if the image contains nature-made or

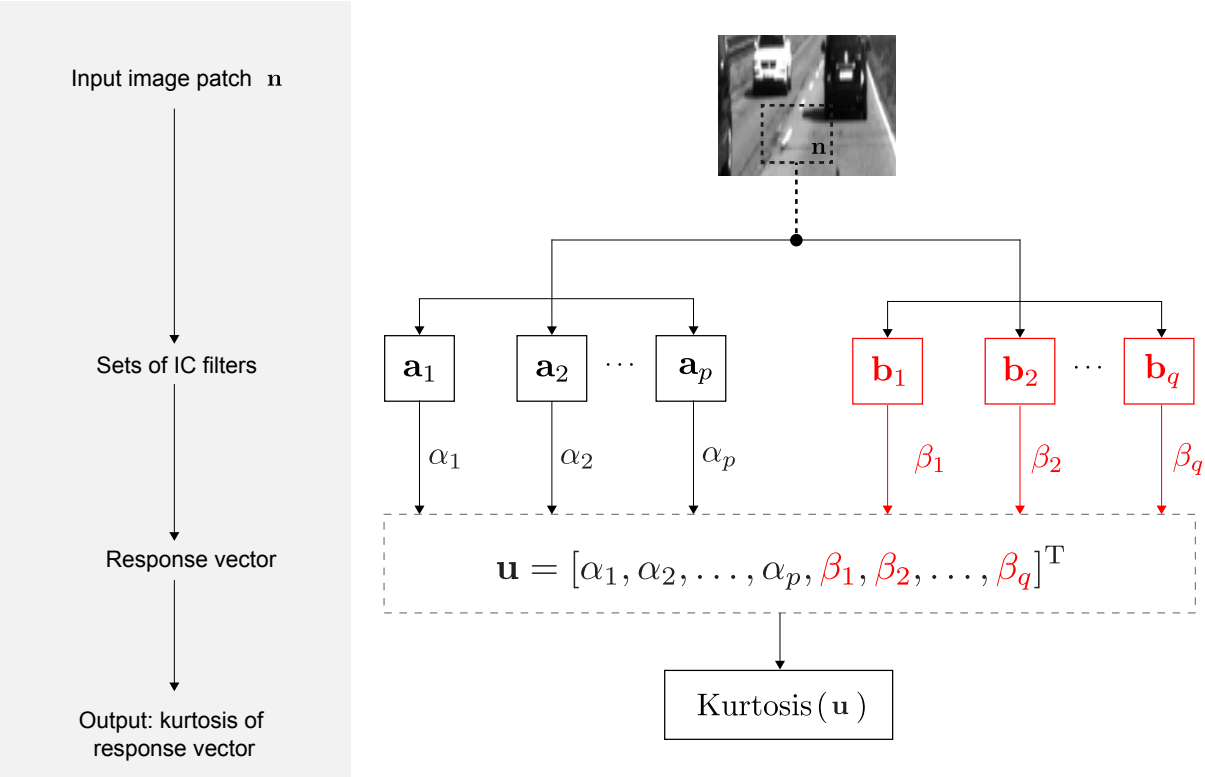

Figure 2.3: **The proposed methodology for visual scene analysis based on the kurtosis of responses of IC filters**. An input image patch is represented by $\mathbf{n}$. Two sets of IC filters are represented by $\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_p$ and $\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_q$, respectively. Notice that $p \gg q$, i.e., the number of filters $\mathbf{a}_i$ is much greater than the number of $\mathbf{b}_i$. The responses of filters $\mathbf{a}_i$ and $\mathbf{b}_i$ to the input image patch $\mathbf{n}$ are represented by α_i and β_i, respectively. The vector $\mathbf{u}$ contain all responses. Finally, the kurtosis of $\mathbf{u}$ is calculated.

man-made structures.

Figure 2.3 shows the block diagram of our methodology. Assume that vector $\mathbf{n} \in \mathbb{R}^n$ represents an input image (this image exhibits either characteristic "A" or "B"). Vectors $\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_p \in \mathbb{R}^n$ and $\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_q \in \mathbb{R}^n$ represent two sets of filters learned by ICA. Also, assume that $p \gg q$, i.e., the number of filters $\mathbf{a}_i$ is much greater than the number of filters $\mathbf{b}_i$. This assumption is necessary because our method uses the kurtosis of filter responses. This is further explained and illustrated below.

Assume that filters $\mathbf{a}_i$ share characteristic "A". For example, if $A = 90$ degrees, then all filters $\mathbf{a}_i$ have spatial orientation equal to 90 degrees. Similarly, if $A =$ high frequency, then all filters $\mathbf{a}_i$ are centered at high frequencies. On the other hand, if $A =$ nature-made, then all filters $\mathbf{a}_i$ are learned from natural scenes.

Assume that filters $\mathbf{b}_i$ share characteristic "B". For example, if $A =$ nature-made and $B =$ man-made, then filters $\mathbf{a}_i$ were learned from nature scenes, and filters $\mathbf{b}_i$ are learned from man-made scenes.

Responses of filters $\mathbf{a}_i$ and $\mathbf{b}_i$ to the input image $\mathbf{n}$ are calculated as

$$\alpha_i = \mathbf{a}_i^{\mathrm{T}} \mathbf{n}, \tag{2.17}$$

$$\beta_i = \mathbf{b}_i^{\mathrm{T}} \mathbf{n}. \tag{2.18}$$

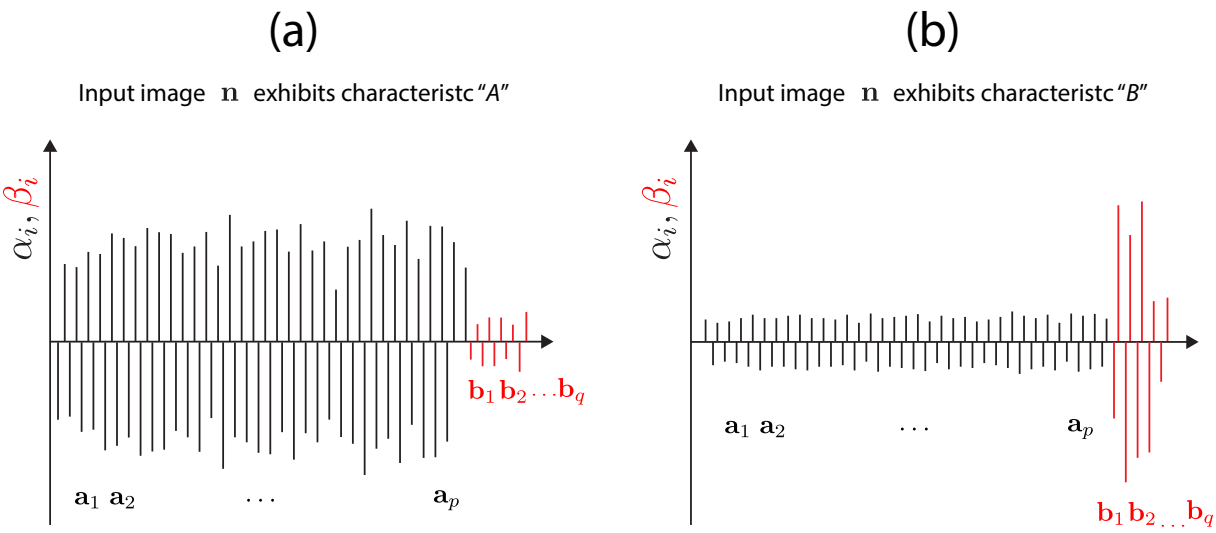

Figure 2.4: **Response of IC filters to an input image patch n**. For (a) and (b), vertical axis represent amplitude of responses α_i (black) and β_i (red). Horizontal axis represent filter name and index, i.e., $\mathbf{a}_i$ (black) or $\mathbf{b}_i$ (red). (a) Input image patch $\mathbf{n}$ and filters $\mathbf{a}_i$ share the same characteristic "A". In this case, responses α_i have higher magnitude than that of β_i. (b) Input image patch $\mathbf{n}$ and filters $\mathbf{b}_i$ share characteristic "B". In this case, β_i have higher magnitude than that of α_i. Notice that since $p \gg q$, the number of responses α_i is much greater than the number of β_i. This can be observed by comparing the number of black and red responses.

Let vector

$$\mathbf{u} = [\alpha_1, \alpha_2, \ldots, \alpha_p, \beta_1, \beta_2, \ldots, \beta_q]^{\mathrm{T}} \tag{2.19}$$

contain the all responses α_i and β_i. For a simpler notation, let us rewrite $\mathbf{u}$ as

$$\mathbf{u} = [u_1, u_2, \ldots, u_k]^{\mathrm{T}}, \tag{2.20}$$

where $u_i = \alpha_i$ for $i \leq p$, $u_i = \beta_{i-p}$ for $i > p$, and $k = p + q$. Finally, the output of the methodology is the kurtosis of vector $\mathbf{u}$. This kurtosis is calculated as

$$K_u = \frac{\frac{1}{k}\sum_{i=1}^{k}[u_i - \mu_u]^4}{\left\{\frac{1}{k}\sum_{i=1}^{k}[u_i - \mu_u]^2\right\}^2}, \tag{2.21}$$

where μ_u is the sample mean of $\mathbf{u}$. This completes the methodology description.

Now, our hypothesis is as follows. If input $\mathbf{n}$ and filters $\mathbf{a}_i$ share characteristic "A", then responses α_i are more likely to have higher magnitude than those of β_i. On the other hand, if input $\mathbf{n}$ and filters $\mathbf{b}_i$ share characteristic "B", then responses β_i are more likely to have higher magnitude than those of α_i. This is illustrated in Figure 2.4. In Figure 2.4(a), responses α_i have magnitude higher than that of β_i. In Figure 2.4(b), responses β_i have magnitude higher than that of α_i.

Since $p \gg q$, the number of responses α_i is much greater than the number of β_i. Thus, only few responses have high magnitude in Figure 2.4(b). In our hypothesis, this effect is similar to that observed for numbers generated from Student's t distribution in

Figure 2.2(b). For this reason, we use kurtosis of $\mathbf{u}$ to discriminate or distinguish the response pattern in Figure 2.4(b) from that in Figure 2.4(a). Specifically, we expect that the response pattern in Figure 2.4(b) have higher kurtosis than that in Figure 2.4(a). In this way, the kurtosis of responses of IC filters is used to detect or recognize if input image patch $\mathbf{n}$ exhibits either a characteristic "A" or a characteristic "B".

Chapter 3

Segmentation of depth-of-field (DOF) image

3.1 Introduction

Generally, segmentation of depth-of-field (DOF) image is achieved by two approaches: edge-based (or boundary detection) and region-based methods. For edge-based methods such as [56], the first step is to detect object boundaries based on the local characteristics of image edges. Then, the focused object is segmented based on an edge-linking procedure.

In region-based methods, DOF segmentation is achieved by analyzing the local high-frequency content of the scene. Specifically, focused areas generally have higher energy in high-frequency than at low-frequency. Therefore, region-based methods usually work by quantifying the difference between the energy in high-frequency and the energy in low-frequency in the image. One example of region-based method is to calculate the energy of responses of wavelet filters centered at high-frequencies [57]. Whenever an image regions presents low energy in high-frequencies, these filters will exhibit responses of low magnitude.

A region-based method has also been proposed based on local (non-normalized) fourth-order moment [58]. Specifically, a map is computed based on the block-wise fourth-order moment of pixel intensities. This map is further processed by morphological operations in order to fill holes and exclude image "blobs". This approach has been improved in [59] by applying further bilateral filtering.

Another method firstly detects edges, then classify image regions into focused object or unfocused background based on a type of fuzzy membership degree [60]. Another approach is proposed based on five stages [61]. These stages are as follows. Firstly, "sharp" pixels are detected. Secondly, these pixels are clustered together. In the third stage, a binary mask is generated by connecting clusters into a contiguous area. The forth stages consists of color segmentation. Finally, pixels in the binary mask are removed or not based on the color segmented groups.

Notice that there are also hybrid approaches formed by combining elements from edge- and region-based methods. For instance, [17] firstly creates an edge map by applying Sobel operator on the input image. Furthermore, it calculates the local kurtosis map from [58]. Then, it generates a third map by dividing the each value in the edge map by the maximum value of the kurtosis map. This process is then repeated for a low-pass filtered version of the input image. In this way, two maps are generated, one for the original image and one for the low-pass image. These two maps are then combined into a

unique map by subtracting the first from the latter. The resulting map is then processed by a series of segmentation processes to generate the final result.

Another hybrid approach has been recently proposed in [62]. This method firstly applies a Gabor filter to the input image and calculates its output. Secondly, it creates an initial segmented map by subtracting the original image from the Gabor filtered image. The resulting map is processed by a method called curve evolution or active contour model. The goal of active contour model is to find the boundaries of the focused object in the depth-of-field image. In active contour model, inner pixels (i.e., pixels within the focused object) and outer pixels (i.e., pixels external to the focused object) are described by different probability distributions. By calculating the parameters of these distributions, the method is able to segment the boundaries pixels from the rest of image pixels. After this first stage, the active contour model in [62] employs the Chan-Vese energy function [63] for a secondary segmentation of boundaries pixels. The final result is computed by summing the segmented image from probability distribution estimation process and the segmented image generated by Chan-Vese energy function.

It is also important to discuss research on general segmentation of objects of interest. Specifically, objects of interest are not always determined by depth-of-field. For instance, a method has been recently proposed to detect image regions which are perceptually salient [64]. That method uses a Gaussian mixture model to decompose the input image in different components. The method then computes the probability of image pixels belonging to the each of those components. The segmented map is generated by using only components of high probability in the Gaussian mixture model.

Another proposed work creates a window which slides over the input image [65]. Image characteristics such as pixel intensity, contrast, color and motion information are firstly computed for pixels internal to the window. Then, these image characteristics are computed for pixels external to the window. Given the image characteristics values, the method calculates a conditional probability distribution of image pixels being internal to the window. Given the image characteristics values, it also calculates the conditional probability distributions of pixels being external to the window. These conditional probability distributions are approximated simply as low-pass filtered histograms of the image characteristics for internal and external pixels. In that method, the conditional probabilities of internal pixels are considered as saliency values. The method then finally uses a conditional random field to determine which image pixels have maximum saliency.

In regard of saliency detection, a method proposes to measure saliency in the CIELAB color space [66]. That method computes the CIELAB space representation of the input image. Also, it computes the CIELAB space representation of a Gabor filtered version of the original image. The saliency map is generated by computing the local or block-wise difference between the two CIELAB representations. Another method for saliency detection is the famous Koch's visual attention model [14,15] previously discussed in the introductory chapter.

Applications

The introductory chapter of this thesis describe several applications of DOF segmentation. For instance, the use of DOF segmentation for enhancing microscopy image profiles [18, 19]. This section describes the characteristics of methods for this application.

Generally, the enhancing of microscopy image profiles is based on a technique known as *extended depth-of-field*. In extended depth-of-field, several images of the same scene are

taken but with different DOF. These different DOF images are then combined into a single image in which all regions are (maximally) in focus. The combination process is generally performed in wavelet domain rather than in the spatial domain [18]. The combination process consists of selecting wavelet components which have high contribution to focused pixels, and excluding wavelet components which have low contribution to focused pixels. This contribution is quantified in terms of pre-defined measures of focus. These pre-defined measures of focus represent the depth-of-field of wavelet components rather than that of pixels in the image.

Open problems

One of the main open problems of methods for DOF segmentation is time consumption. For instance, the very recently proposed active contour approach for DOF segmentation [62] has one of the highest accuracy to date. However, it can be more extremely slow in comparison to other methods. This may not be satisfactory for a real-world application. Furthermore, accuracy of the active contour approach largely varies over different image sizes.

Another problem is computational complexity. For instance, methods which rely on many stages of segmentation or on probability estimation algorithms may be very difficult to implement in hardware platforms used for image segmentation in real-time industrial and consumer applications.

3.2 DOF segmentation method

Our method for DOF segmentation is based on the methodology proposed in section 2.2. This methodology requires the definition of characteristics "A" and "B". Furthermore, the methodology requires IC filters $\mathbf{a}_i$ and $\mathbf{b}_i$ which exhibit characteristics "A" and "B", respectively.

Since DOF segmentation consists of discriminating unfocused and focused images areas, characteristics "A" and "B" could be chosen as A = unfocused and B = focused. In this way, IC filters $\mathbf{a}_i$ could be learned from unfocused images, and filters $\mathbf{b}_i$ could be learned from focused images. We use a simpler approach. It is noticed that focused images have higher energy in high frequencies than that of unfocused images. Thus, we choose characteristics "A" and "B" as A = high-frequency and B = low-frequency. In this way, filters $\mathbf{a}_i$ should be centered at high-frequencies, and filters $\mathbf{b}_i$ should be centered at low-frequencies. The next section describes how filters $\mathbf{a}_i$ and $\mathbf{b}_i$ are learned. The block diagram of our method for DOF segmentation is shown in Figure 3.1.

In the first step, the RGB color bands of the input image are collapsed generating a grayscale image $\mathbf{I}$. Around a pixel $I_{x,y}$ of this grayscale image, let us consider a neighborhood of $2L+1 \times 2L+1$ pixels. This neighborhood is represented by the column vector $\mathbf{n}_{x,y}$. This vector is created by reading the pixels in the neighborhood in a column-wise fashion, i.e., from top to bottom and left to right.

The second step consists of log-transforming luminance values in the neighborhood, i.e.,

$$\mathbf{n}'_{x,y} = \log^*(\mathbf{n}_{x,y}), \tag{3.1}$$

where $\log^*(a) = 0$ for $a < 1$ and $\log^*(a) = \log(a)$ for $a \geq 1$. This non-linear transformation reduces large differences between luminance intensities in different parts of the

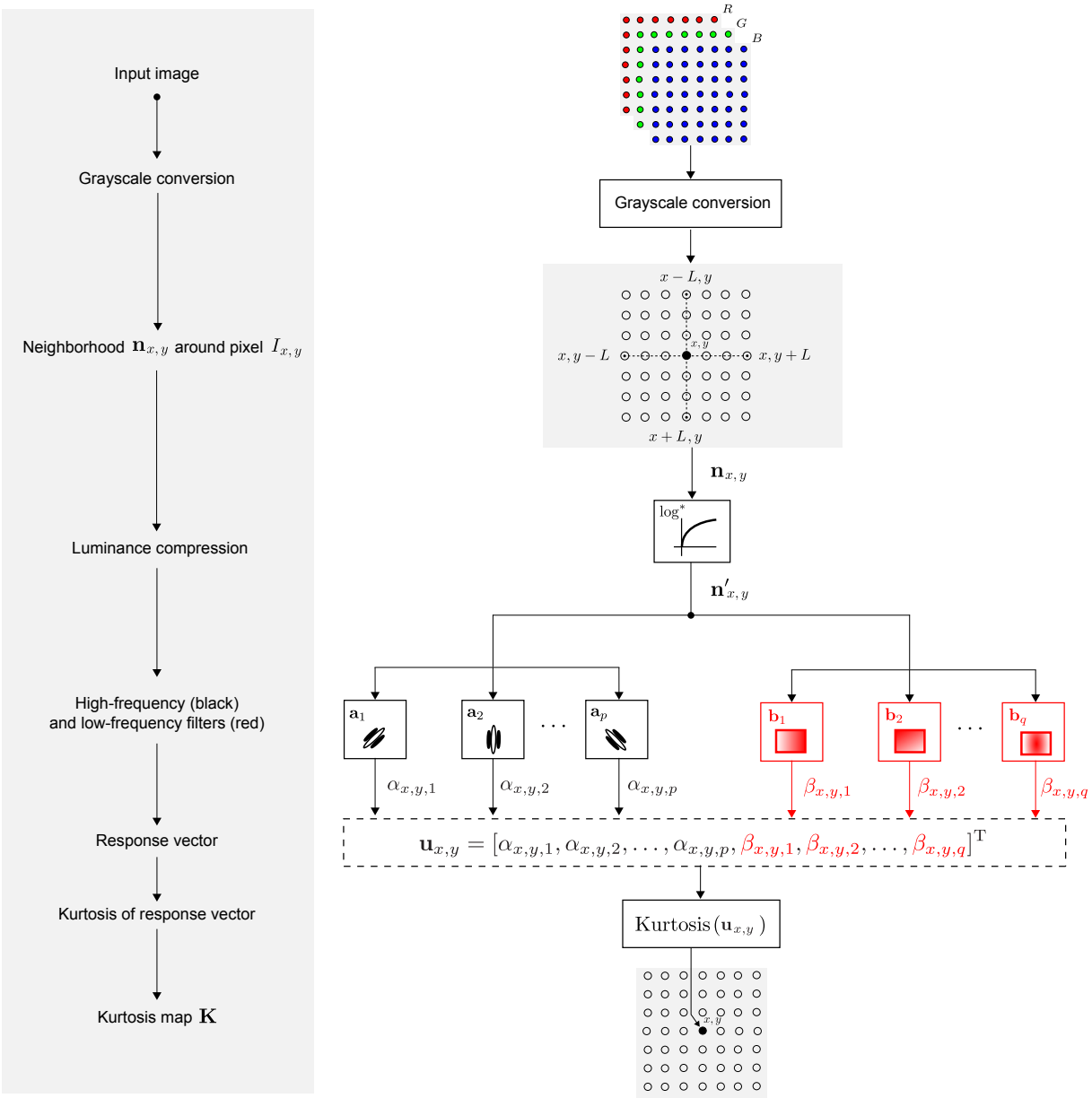

Figure 3.1: **Overview for proposed DOF segmentation.** An input image is converted into a grayscale image $\mathbf{I}$. Around a pixel $I_{x,y}$ of this grayscale image, consider a neighborhood of pixels represented by $\mathbf{n}_{x,y}$, where (x, y) represents coordinates of a pixel. The luminance of pixels are log-transformed generating $\mathbf{n}'$. Two sets of IC filters are represented by $\mathbf{a}_i$ and $\mathbf{b}_i$. Notice that $p \gg q$. Filters $\mathbf{a}_i$ are centered at high frequencies. Filters $\mathbf{b}_i$ are centered at low frequencies. The responses of filters $\mathbf{a}_i$ and $\mathbf{b}_i$ are represented by $\alpha_{x,y,i}$ and $\beta_{x,y,i}$, respectively. Vector $\mathbf{u}_{x,y}$ contain both responses $\alpha_{x,y,i}$ and $\beta_{x,y,i}$. The kurtosis map $\mathbf{K}$ is generated by computing the kurtosis of $\mathbf{u}_{x,y}$ for every (x, y).

image.

After luminance normalization, the responses of filters $\mathbf{a}_i$ and $\mathbf{b}_i$ are calculated as

$$\alpha_{x,y,i} = \mathbf{a}_i^{\mathrm{T}} \mathbf{n}'_{x,y}, \tag{3.2}$$

$$\beta_{x,y,i} = \mathbf{b}_i^{\mathrm{T}} \mathbf{n}'_{x,y}, \tag{3.3}$$

respectively. Let a vector

$$\mathbf{u}_{x,y} = [\alpha_{x,y,1}, \alpha_{x,y,2}, \ldots, \alpha_{x,y,p}, \beta_{x,y,1}, \beta_{x,y,2}, \ldots, \beta_{x,y,q}]^{\mathrm{T}} \tag{3.4}$$

contain the all responses $\alpha_{x,y,i}$ and $\beta_{x,y,i}$. For an simpler notation, let us rewrite $\mathbf{u}_{x,y}$ as

$$\mathbf{u}_{x,y} = [u_{x,y,1}, u_{x,y,2}, \ldots, u_{x,y,k}]^{\mathrm{T}}, \tag{3.5}$$

where $u_{x,y,i} = \alpha_{x,y,i}$ for $i \leq p$, $u_{x,y,i} = \beta_{x,y,i-p}$ for $i > p$, and $k = p + q$. Finally, the kurtosis of $\mathbf{u}_{x,y}$ is computed as

$$K_{x,y} = \frac{\frac{1}{k}\sum_{i=1}^{k}[u_{x,y,i} - \mu_u]^4}{\left\{\frac{1}{k}\sum_{i=1}^{k}[u_{x,y,i} - \mu_u]^2\right\}^2}, \tag{3.6}$$

where μ_u is the mean value of vector $\mathbf{u}_{x,y}$. Since kurtosis of $\mathbf{u}_{x,y}$ is calculated for every possible (x, y), values $K_{x,y}$ form a kurtosis map.

Now, let us revisit our hypothesis illustrated in Figure 2.4. When image $\mathbf{n}$ exhibits characteristic "A", the response pattern is similar to Figure 2.4(a). When image $\mathbf{n}$ exhibits characteristic "B", the response pattern is similar to Figure 2.4(b). Finally, the kurtosis of the response pattern in Figure 2.4(a) is expected to be higher than that of Figure 2.4(b).

Applying our hypothesis for the problem of DOF segmentation, we have that characteristic A = high-frequency and B = low-frequency. Therefore, if input image $\mathbf{n}_{x,y}$ has low energy in high frequencies, kurtosis of $\mathbf{u}_{x,y}$ is expected to be high. If input image $\mathbf{n}_{x,y}$ has high energy in high frequencies, kurtosis of $\mathbf{u}_{x,y}$ is expected to be low. This is verified in the section *Experiments and results*.

Generating final segmented image

After computing the kurtosis map $\mathbf{K}$, it is necessary to generate the final segmented image. This process consists of two steps, i.e., *morphological reconstruction* and *thresholding*. The goal of morphological reconstruction is to fill "holes" inside $\mathbf{K}$. This process is as follows:

1. **Define**

$$J_{x,y} = \begin{cases} -K_{x,y} & \text{if } x, y \text{ is on the map borders,} \\ M & \text{otherwise,} \end{cases}$$

where $M < -K_{x,y}\ \forall x, y$. Here, it is used $M = \min_{x,y}\{-K_{x,y}\} - 1$.

2. **While** $J_{z+1} \neq J_z$ **do**

$$J_{x,y} \leftarrow \min\{[J \oplus B]_{x,y}, -K_{x,y}\}, \tag{3.7}$$

where $[J \oplus B]_{x,y} = \max_{(x',y') \in N_B} \{J(x - x', y - y')\}$ is the dilation of matrix $\mathbf{J}$ by a flat structuring element $\mathbf{B}$. Here, it is used a 3×3 flat element so that N_B is an 8-connectivity neighborhood.
end of while.
3. **Return** $\mathbf{J}^*$.

After this procedure is executed, the resulting map after filling holes given by $\mathbf{J}^*$. The goal of the second step, i.e., thresholding, is to binarize map $\mathbf{J}^*$. Specifically,

$$A_{x,y} = \begin{cases} 1 & \text{if } J^*_{x,y} > T \\ 0 & \text{otherwise,} \end{cases} \tag{3.8}$$

where $T \in \mathbb{R}$ is a threshold, and $\mathbf{A}$ is the binarized version of $\mathbf{J}^*$. T is determined by Otsu's method [67]. Ostu's method chooses the threshold so as to minimize the intraclass variance of 0 and 1 values. Finally, the segmented image is found by multiplying the original image $\mathbf{I}$ by the binary image $\mathbf{A}$.

3.3 Experiment and results

Learned IC filters

This section describes how IC filters $\mathbf{a}_i$ and $\mathbf{b}_i$ are learned. In our model for DOF segmentation, filters $\mathbf{a}_i$ must be centered at high-frequencies, and filters $\mathbf{b}_i$ must be centered at low-frequencies. Furthermore, the number of filters $\mathbf{a}_i$ must be much greater than that of filters $\mathbf{b}_i$.

In order to learn IC filters, a dataset of nature scenes was obtained from the McGill Calibrated Color Image Database [68]. From this database, we have randomly selected 100 natural scenes. From these images, 100,000 patches of 16 x 16 pixels were extracted in a non-overlapping fashion. This set of image patches was used as input for the FastICA algorithm. Figure 3.2 exhibits the learned IC filters.

Notice that IC filters learned from natural images are visually similar to two-dimensional Gabor functions. A two-dimensional Gabor function is defined as the product between a Gaussian envelop and a cosine grating, i.e.,

$$g(x, y) = e^{\frac{1}{2}\left(-\frac{x'^2}{\sigma_x} - \frac{y'^2}{\sigma_y}\right)} \cos(2\pi f x' + \phi), \tag{3.9}$$

where

$$\begin{aligned} x' &= (x - x_c)\cos\theta + (y - y_c)\sin\theta, \\ y' &= -(x - x_c)\sin\theta + (y - y_c)\cos\theta, \end{aligned} \tag{3.10}$$

and σ_x, σ_y represent horizontal and vertical lengths. f, ϕ are center frequency and phase. And x_c, y_c are the center position coordinates. The orientation θ determines the direction of oscillation of the cosine component.

In order to quantify the characteristics of IC filters in Figure 3.2, we have approximated each filter by a Gabor function using a fitting algorithm. The fitting algorithm works by minimizing the following error

$$\min_{x_c, y_c, \sigma_x, \sigma_y, f, \phi, \theta} \|\mathbf{w}_i - \mathbf{g}_i\|_2, \tag{3.11}$$

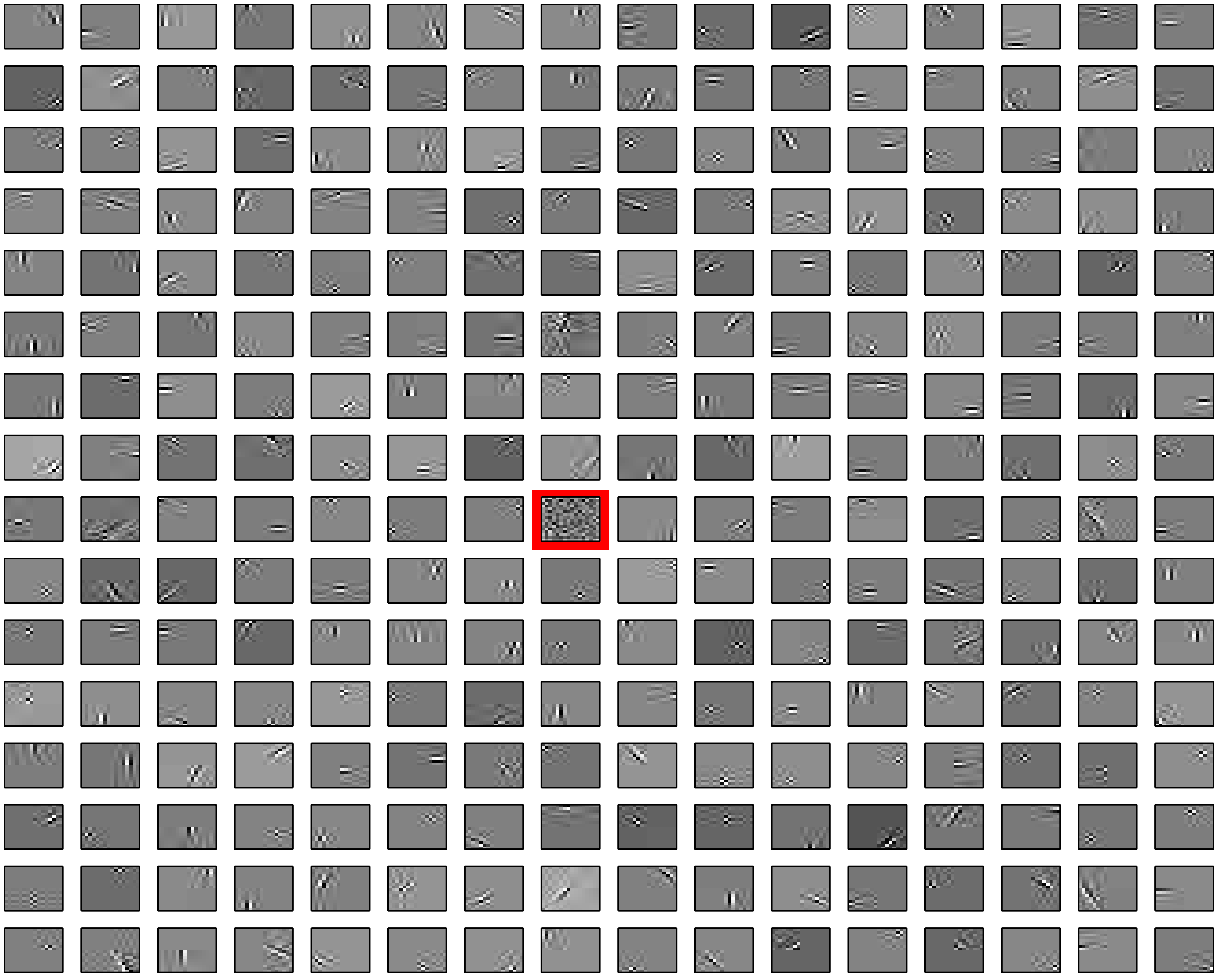

Figure 3.2: **Independent component filters learned from natural images**. There are 256 IC filters exhibited in the figure. Each filter is represented by a square of dimensions 16 x 16 pixels. This bank of IC filters has been learned when the input variables were pixels of natural images. The ICA algorithm used for learning was the FastICA algorithm [69]. The filter surrounded by red line is the direct-component filter.

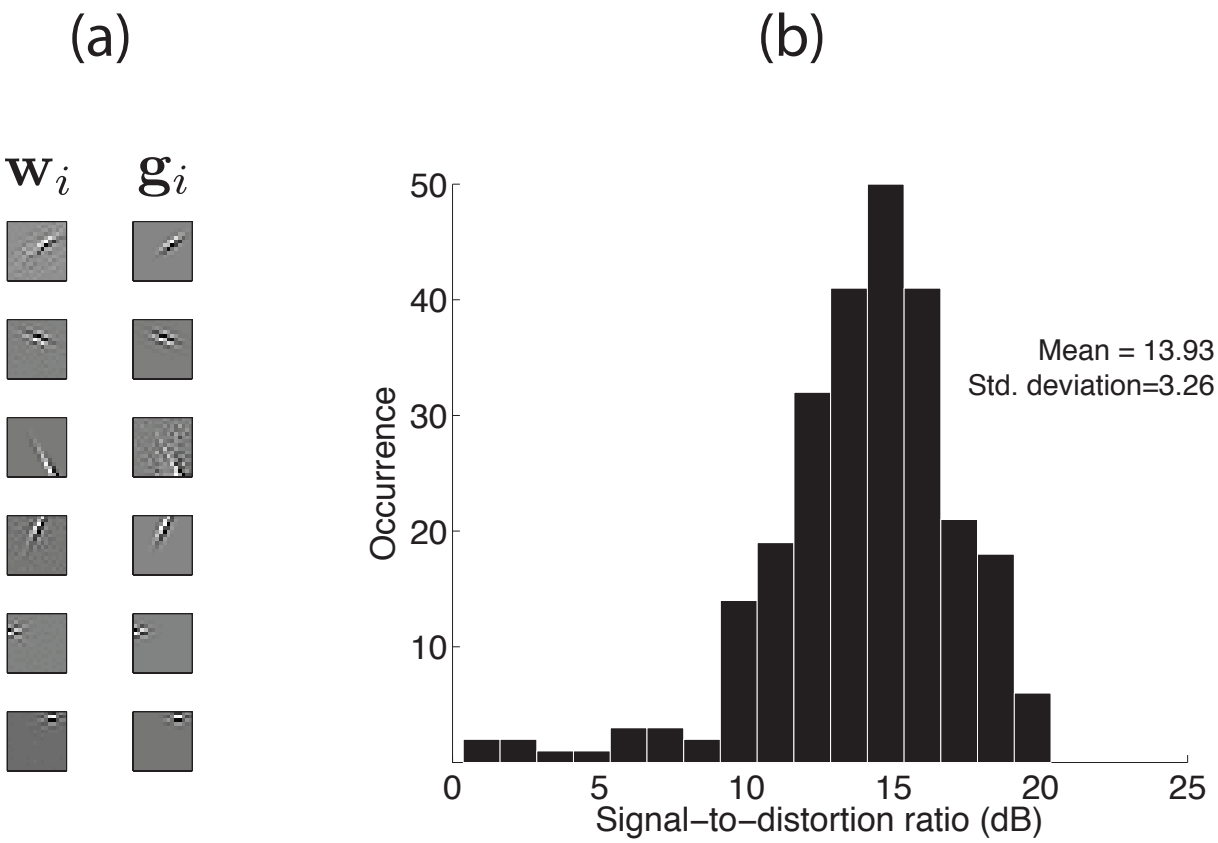

Figure 3.3: **IC filters and estimated Gabor functions**. For each IC filter in Figure 3.2, we have estimated a Gabor function by using a fitting algorithm. This algorithm works by minimizing the error in Eq. (3.11). (a) Examples of IC filters $\mathbf{w}_i$ and the estimated Gabor functions $\mathbf{g}_i$. (b) Histogram of signal-to-distortion ratio (SDR) between IC filters and Gabor functions, i.e., $SDR = 10 \log_{10} \frac{\|\mathbf{w}_i\|_2}{\|\mathbf{w}_i - \mathbf{g}_i\|_2}$. The statistics of the histogram are also shown, i.e., mean, standard deviation, skewness and kurtosis. The high SDR values suggests that Gabor functions are good approximations of IC filters.

where vectors $\mathbf{w}_i$ and $\mathbf{g}_i$ represent the IC filter and the estimated Gabor function $g(x, y)$, respectively. Vector $\mathbf{g}_i$ is obtained by reading values of the two-dimensional function $g(x, y)$ in raster scan fashion, i.e, from left to right and top to bottom.

Figure 3.3(a) shows examples IC filters and fitted Gabor functions. Figure 3.3(b) shows the histogram of signal-to-distortion ratio (SDR) between IC filters and fitted Gabor functions. The histogram shows that the distortion between the majority of IC filters and Gabor functions is between 10 dB and 20 dB.

The high SDR (i.e., $SDR = 10 \log_{10} \frac{\|\mathbf{w}_i\|_2}{\|\mathbf{w}_i - \mathbf{g}_i\|_2}$) suggests that Gabor functions are good approximations of IC filters. Therefore, we can use the parameters of Gabor functions to characterize the IC filters. These parameters are center frequency f, orientation θ and the area in the frequency domain approximated by $\frac{1}{\sigma_x \sigma_y}$. These parameters are shown in Figure 3.4.

In the polar plot in Figure 3.4(a), each circle represents an IC filter. The distance of the circle to the origin of the plot represents center frequency f (given in cycles per pixel). The orientation of the circle (given in degrees) represents orientation θ. The color of the circle represents the area in frequency domain occupied by the IC filter. The respective

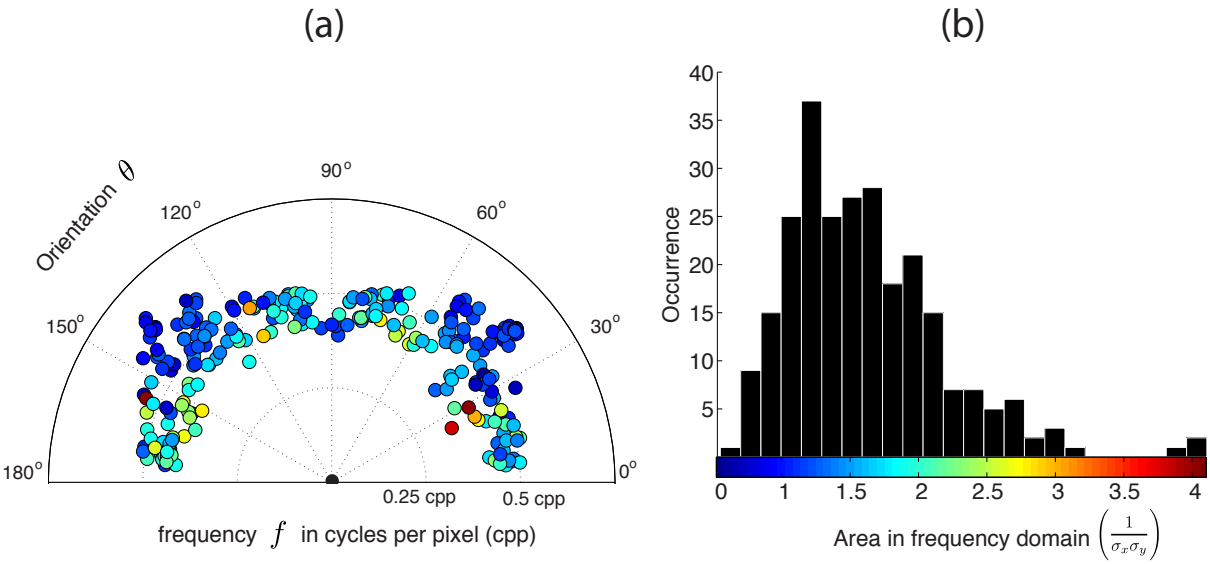

Figure 3.4: **Estimated parameters for the IC filters in Figure 3.2**. (a) In the polar plot, each circle represents an IC filter. The distance of the circle to the origin of the plot represents the frequency f (given in cycles per pixel). The orientation of the circle (given in degrees) represents the orientation θ. The color of the circle represents the area in frequency domain occupied by the IC filter. (b) The respective colormap and histogram for area in frequency domain. The polar plot shows that the IC filters are centered at high frequencies. Furthermore, the number of filters close to oblique orientations ($\theta = 45$ degrees and $\theta = 135$ degrees) is higher than that at normal orientations ($\theta = 0$ degrees and $\theta = 90 degrees$). According to the color of circles, however, filters at oblique orientations have smaller area in the frequency domain. Only a few number of filters have larger areas in the frequency domain.

colormap and histogram for area in frequency domain is shown in Figure 3.4(b).

The polar plot shows that 255 IC filters are centered at high frequencies. The FastICA algorithm also learned one very-low frequency filter, which is generally named as *direct-component* filter. This filter is highlighted in Figure 3.2 by a red square.

In Figure 3.4(b), notice that the number of filters close to oblique orientations ($\theta = 45$ degrees and $\theta = 135$ degrees) is higher than that at normal orientations ($\theta = 0$ degrees and $\theta = 90$ degrees). According to the color of circles, however, filters at oblique orientations have smaller area in the frequency domain. Notice that the histogram in Figure 3.4(b) shows a high concentration of small areas in frequency domain. Only a few number of filters have larger areas in the frequency domain. In our paper [70], we demonstrate how to control the characteristics of IC filters such as center frequency.

Notice that these IC filters match the requirements for our DOF segmentation method. Specifically, the learned set of IC filters consists of 255 high-frequency filters and one low-frequency filter. Therefore, we use those filters as the required filters $\mathbf{a}_i$ and $\mathbf{b}_i$, respectively. In this way, there are 255 filters $\mathbf{a}_i$ and one filter $\mathbf{b}_i$.

Differences between IC filters and Gabor functions

Before presenting results for DOF segmentation, let us discuss the following issue. Although Gabor functions are good approximations of IC filters, there are differences between them [71]. In order to analyze these differences, we compute the average 2D Fourier amplitude spectrum of IC filters, and the average 2D Fourier amplitude spectrum of Gabor functions. These spectra are shown in Figure 3.5.

In both Figures 3.5(a) and 3.5(b), for a fixed spatial frequency, amplitude is higher at oblique orientations (45 degrees and 135 degrees) than at normal orientations (0 degrees and 90 degrees). Now, let us analyze the "isolines" or profile curves of the amplitude spectra. Specifically, let's analyze isolines for low spatial frequencies, i.e., frequencies close the origin of the Fourier domain ($u = 0, v = 0$). For the average amplitude spectrum of IC filters (Figure 3.5(a)), isolines have an almost perfect "diamond" shape. For the average amplitude spectrum of Gabor functions (Figure 3.5(b)), isolines have a more circular shape. This difference between IC filters and Gabor functions is an important issue and deserves a discussion on its own. This discussion is presented in Appendix C.

Results for DOF segmentation

In order to demonstrate how kurtosis of IC filters represents depth-of-field, we have chosen two images from the famous Berkeley Segmentation Dataset and Benchmark500 (BSDS500) [72]. These images are shown in Figure 3.6(a). The kurtosis maps generated by our methodology are shown in Figure 3.6(b). As expected, kurtosis is higher for unfocused areas. Figure 3.6(c) shows the segmented images obtained from the kurtosis maps in Figure 3.6(b).

For the rest of this section, two image datasets are used for evaluation. The first dataset consists of 86 DOF images selected from the Berkeley Segmentation Dataset and Benchmark500 (BSDS500). These images are encoded in JPEG format. Their size is 321×481 pixels. The second dataset is the GraKriWei Dataset [61]. This dataset consists of 63 DOF images. These images are encoded in a very high quality JPEG format. The sizes of these images are very large and not identical.

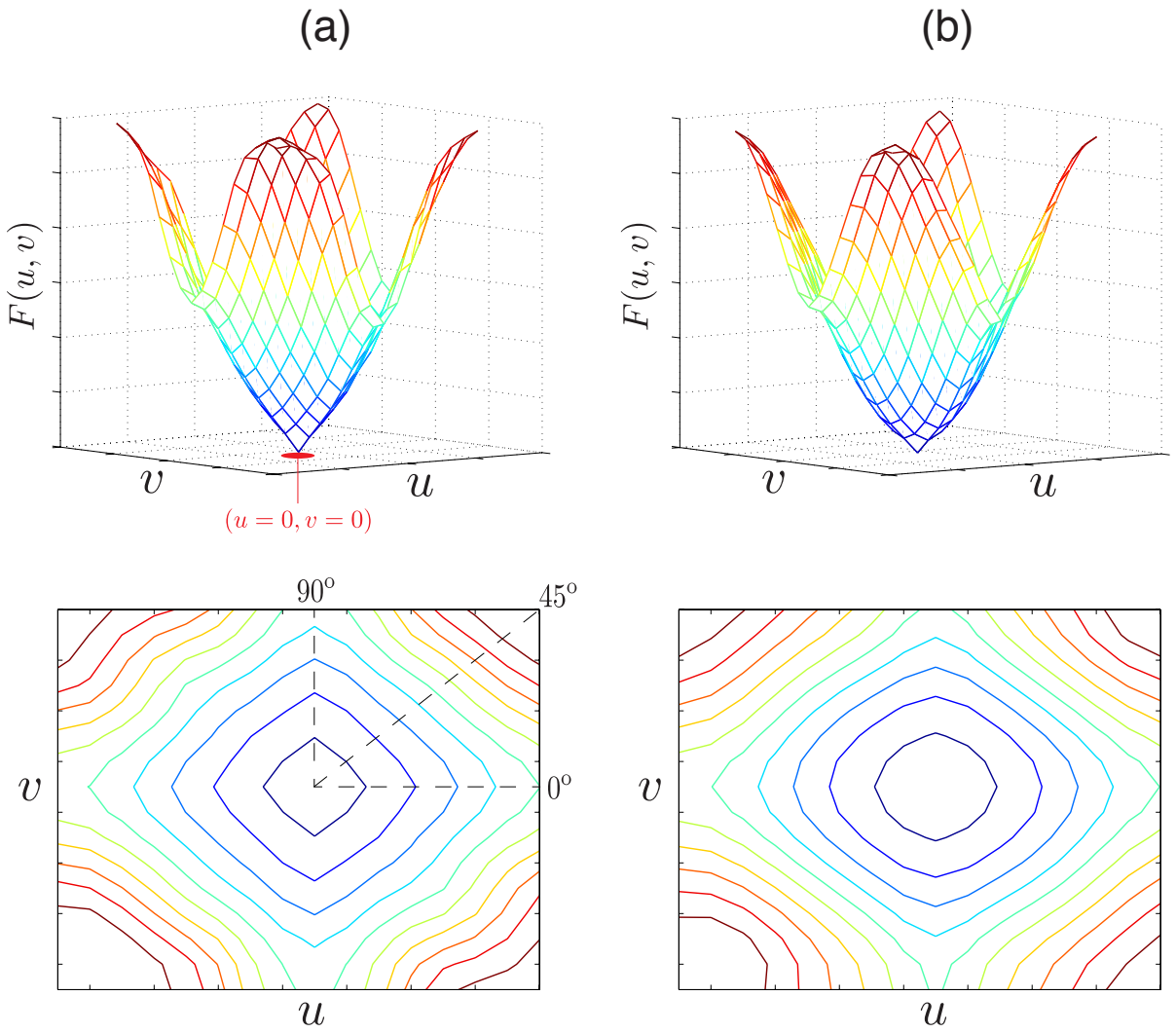

Figure 3.5: **Average 2D Fourier amplitude spectra**. (a) Average amplitude spectrum of IC filters. (b) Average amplitude spectrum of Gabor functions. The origin of the Fourier domain is represented by the point $(u = 0, v = 0)$. The plots at the bottom show the profile curves or "isolines" for different levels of amplitude. Notice that for a fixed spatial frequency, amplitude is higher at oblique orientations (45 degrees and 135 degrees) than at normal orientations (0 degrees and 90 degrees).

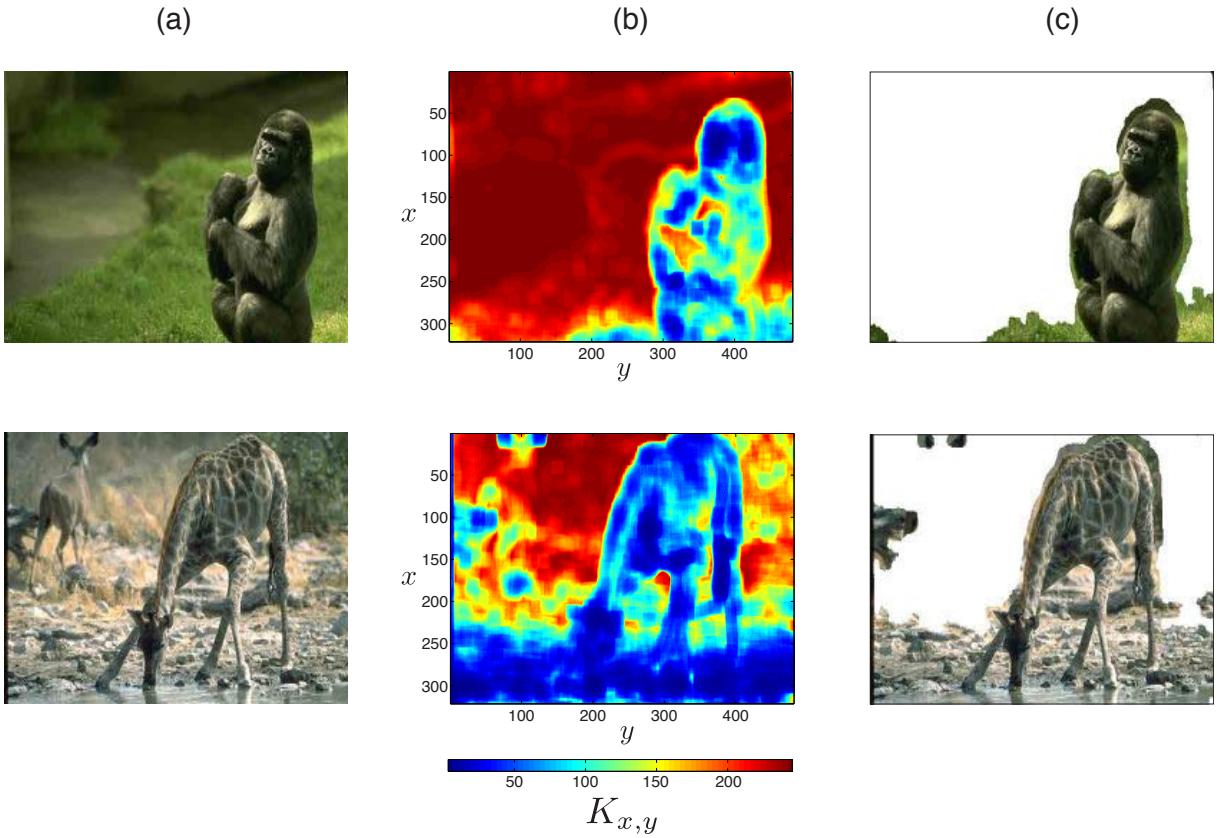

Figure 3.6: **Segmentation of depth of field by using kurtosis of responses of IC filters**. (a) Original images. (b) Kurtosis maps. Unfocused regions exhibit higher Kurtosis $K_{x,y}$ than that of focused regions. (c) Segmented images.

Table 3.1: **Segmentation performance on Berkeley Segmentation Dataset and Benchmark500**.

	Fuzzy [60]	NonNorm4 [59]	Proposed method [73]	Score [61]	Cur.Evo [62]
Average of F-values	0.29	0.68	0.72	0.77	0.85
Standard deviation of F-values	0.28	0.21	0.20	0.16	0.11
Average processing time	46 s	3 s	6 s	73 s	241 s

Table 3.2: **Segmentation performance on GraKriWei Dataset**.

	Fuzzy [60]	NonNorm4 [59]	Proposed method [73]	Score [61]	Cur.Evo. [62]
Average of F-values	0.38	0.61	0.71	0.83	0.86
Standard deviation of F-values	0.25	0.21	0.18	0.18	0.11
Average processing time	51 s	3 s	6 s	79 s	249 s

For all these images, ground-truth masks are provided. These masks are binary images in which pixels with values "0" and "1" indicate unfocused and focused pixels in the original image, respectively. In this way, ground-truth masks are used to determine which pixels are correctly or wrongly classified in segmented images. For these two image datasets, the performance of recently proposed methods for DOF segmentation [62] has been quantified using the F-measure. Therefore, we also use the F-measure. The F-measure is defined as

$$F = 2\frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}. \tag{3.12}$$

Precision and recall are computed as

$$\begin{aligned} \text{precision} &= \frac{tp}{tp+fp}, \\ \text{recall} &= \frac{tp}{tp+fn}, \end{aligned} \tag{3.13}$$

where tp is the number of focused pixels correctly classified as focused, fp is the number of unfocused pixels wrongly classified as focused, and fn is the number of focused pixels wrongly classified as unfocused. In general, larger F-values indicate larger number of correctly segmented pixels.

Tables 3.1 and 3.2 show the results for Berkeley Segmentation Dataset and GraKriWei Dataset, respectively. In these tables, columns indicate different methods. The first row shows the average of F-values over the images in the dataset. The second row shows the standard deviation of F-values. The last row shows the average time required to process an image.

In Tables 3.1 and 3.2, the second and third columns are the Zhang's fuzzy segmentation approach [60] and Li's method based on the non-normalized fourth-order moment [59], respectively. The fourth column is the proposed method. The fifth column is Graf's method based on score region and color segmentation [61]. The last column is the active contour approach proposed by Mei et al. [62].

From Tables 3.1 and 3.2, it is said that our method has competitive performance in terms of average F-value and time consumption. Notice that the methods with higher average F-value are far slower than our method. The method with highest average F-value is the curve evolution approach described in [62]. Notice, however, that this method is the most time consuming. It is important to highlight that for the majority of state-of-art methods, time consumption is dependent on the input image. Time consumption for our method is almost constant over different images.

Here, it should be noticed that our method and the active contour methods are executed in Matlab environment. The other methods are executed in compiled software. In this way, a point could be made that our method and the active contour approach could be faster than that reported on Tables 3.1 and 3.2.

For subjective evaluation, Figure 3.7 shows examples of images segmented by our method and by the curve evolution approach with highest performance. For these images, ground-truth masks and objective measures are provided as well.

It is important to discuss the above results in regard of the applications of DOF segmentation cited in the introductory chapter. In this regard, the datasets used to evaluate methods of DOF segmentation consists of general photographs. Thus, the performance reported in our analysis should reflect the performance in applications with general image and video data. However, in relation to applications such as enhancing microscopy images, one should consider the following issues when interpreting the results. For the problem of enhancing microscopy images, the segmentation or processing of depth-of-field is only an intermediate step in the solution. This step is carried out in wavelet domain and not in the space domain for methods based on extended depth-of-field [18]. Furthermore, the final result or output produced in this subject is not a segmented image in which different regions have different depth-of-fields. Rather, the result or output image is an image in which all regions are in focus. In this way, a direct comparison between the performance of methods for enhancing microscopy images and general DOF segmentation is not straightforward.

However, one can compare how extended depth-of-field techniques and general DOF segmentation methods handle depth-of-field. In the extended depth-of-field methodology, one intermediate step calculates depth-of-field contribution of wavelet components based on pre-defined measures of focus. These measures of focus include standard luminance contrast, gradient energy of luminance and the energy of high-frequency components. Although researches on enhancing microscopy images do not generally report objective evaluation of performance for the employed measures of focus, these measures are basically the same criteria used in many methods for general DOF segmentation. In this way, the performance level in our analysis should be satisfactory for application on the subject enhancing microscopy images.

Shortcomings of the proposed method

In order to provide more information about the performance of our method, we show images in which our method fails. There are two main shortcomings in our method for DOF segmentation. The first shortcoming is related to the boundaries of focused objects. The second shortcoming is related to low-frequency image regions which are in focus. Figure 3.8 illustrates these shortcomings.

In Figure 3.8(b), Notice that between the "true boundary" and the "false boundary", our method (wrongly) segments pixels which are not in focus. Furthermore, notice that our method doesn't segment low-frequency regions which are in focus. These problems can also be observed at the results shown in Figure 3.7(c).

Better segmentation of boundaries are actually achieved by active contour methodologies. However, such methods are extremely time consuming. Better segmentation of low-frequency regions in focus is also a difficult problem. The reason is that low-frequency regions generally lack features such as image edges. Thus, algorithms do not have clues or "what" to process. Human perception may solve this problem by using some type

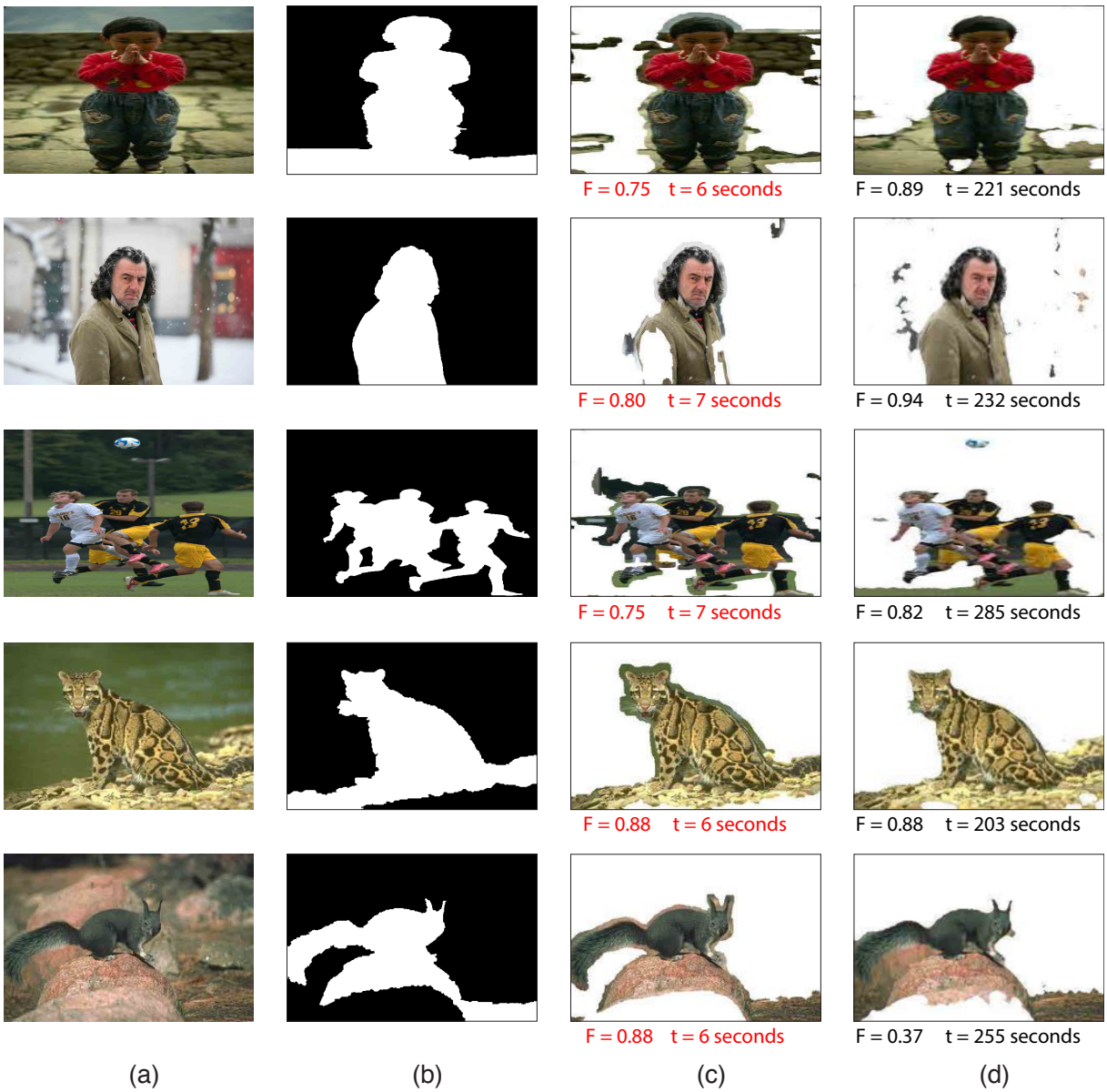

Figure 3.7: **DOF segmentation**. (a) Original images. (b) Ground-truth masks. (c) Proposed method. (d) Active contour approach [62]. F-values and processing time in seconds are shown below each segmented image.

Figure 3.8: **Shortcomings of the proposed method**. (a) Original image. (b) Segmented image. Notice that between the "true boundary" and the "false boundary", our method (wrongly) segments pixels which are not in focus. Furthermore, our method doesn't segment low-frequencies regions which are in focus.

higher-order processing architecture. Such architectures usually consist of different layers of filters. According some defined criteria, filters in higher layers are more complex than those in lower layers. The obvious drawback is that processing time will increase considerably.

3.4 Conclusion

This chapter has presented the method for segmenting depth-of-field images based on our methodology for visual scene analysis. Here, the required filters $\mathbf{a}_i$ and $\mathbf{b}_i$ consists of IC filters centered at high frequencies and low frequencies, respectively. Since filters $\mathbf{b}_i$ are centered at low frequencies, the kurtosis of filter responses is high for low-frequency input images, and it is low for high-frequency input images.

The main advantages of the proposed method are the unsupervised and fast execution. Furthermore, our method is quite easy to implement. All these characteristics are attractive for real-time industrial and consumer applications.

The performance of our method was evaluated on image datasets used by several other works on depth-of-field segmentation. For these datasets, our method exhibits a competitive performance in comparison to the state of art. However, our method is not without shortcomings, i.e., processing of boundaries and low-frequency image regions in focus.

Still, by observing our results, we conclude that our method improves the state of art among fast DOF segmentation methods. Future works must focus on achieving better segmentation of boundaries and low-frequencies regions, but using processing architectures which are fast.

Chapter 4

Segmentation of natural and man-made structures

4.1 Introduction

Much of research on segmentation of natural and man-made structures focus on aerial images. For instance, Anderson applied edge detection on aerial photographs to identify objects such as buildings, cars, statues and sidewalks [74]. Her methodology consists of extracting length and spatial frequency of image edges. Then, objects are recognized based on the levels or amplitude of these characteristics. Notice that many other methods have been proposed for detecting man-made objects in aerial scenes [32, 75–78].

In a different line of research, Torralba et al. proposed to classify visual scenes into two categories, i.e., nature scene or urban scene [79]. Their methodology consists of analyzing the shape of the power spectra of the image. Specifically, the shape of the average power spectra of nature scenes is similar to a "diamond" pattern. On the other hand, the shape of the average power spectra of urban scenes is similar to a "star" pattern. Notice that their methodology is used to classify the entire image. In other words, their methodology does not determine spatial position of man-made structures within the scene.

Caron et al. proposed to determine spatial position of man-made object in a natural scene [80]. Their methodology consists of calculating Zipf's law distribution of image patches. Man-made objects are detected based on the amplitude of Zipf's law distribution. Kumar et al. also proposed to detect man-made structures in scenes [81, 82]. Their methodology consists of using variations of Markov random fields such as multi-scale random fields (MSRF) and discriminative random Fields (DRF) for segmentation and classification of image regions. Vishwanathan et al. proposed a similar approach using conditional random fields (CRF) instead of DRF [83].

We have also previously proposed a method for segmentation of natural and man-made structures [84]. Our previous method uses two image generative models for modeling image regions. In the first model, image patches are modeled as the sum of IC basis functions learned from natural images. In the second model, patches are modeled as the sum of IC basis functions learned from man-made scenes (a matrix containing IC basis functions can be computed as the inverse of a matrix containing IC filters). After modeling, the mean squared errors between original image patches and image patches reconstructed by the two generative models are computed. Image patches are finally classified as nature-made or man-made based on the generative model which generates the smaller error.

Applications

The introductory chapter of this thesis describes several applications of segmentation of natural and man-made structures. For instance, detection of vegetation obstacles during autonomous navigation [20, 21], terrain classification from aerial data to support ground vehicle navigation [22], military applications such as target detection and tracking [32]. This section describes the properties of methods for these applications.

For the detection of vegetation obstacles, [20] proposes a two stage method. The first stage consists of calculating a saliency map of the input scene. This saliency map is calculated based on the product of unidimensional distributions given by the output of independent component filters. The second stage consists of using a multi-class image labeling system for classification of image regions. This image labeling system is based on conditional random fields. [21] also proposes a two stage approach for detecting vegetation in complex driving environments. The first stage consists of calculating a quantity based on near-infrared reflectance known as *normalized difference vegetation index* (NDVI), which varies from -1 (blue sky) to 1 (chlorophyll-rich vegetation). The second stage consists of computing a three-dimensional point distribution in which surface, rectilinear and scattered structures can be discriminated. Based on the output of both stages, a logistic classifier is trained to detect vegetation.

For terrain classification from aerial data to support ground navigation, [22] uses high-density, colorized, three-dimensional laser data. This data is then used as input of a neural network classifier. For target detection and tracking, [32] proposes a method based on the human visual attention mechanism and texture perception. The initial step of that method consists of preprocessing for noise reduction and image contrast enhancement. Then, a stage of feature extraction is started. In this stage, features are extracted to characterize texture. Based on these features for texture characterization, an image segmentation is performed so as to indicate which image regions are likely to contain a target or background. For image regions which likely contain targets, a second feature extraction stage is performed. In this stage, features are extracted to characterize the geometric structure of the object rather than its texture. Based on texture and geometric structure from man-made objects know a priori (e.g., airplanes, tanks, trucks, etc), the system is able to detect targets.

Open problems

There are two problems which all proposed methods have difficulty to handle. The first problem is *generalization*. Specifically, within nature-made or man-made categories, the characteristics of objects can largely vary. For example, a tree is different from the sky or a river. However, these structures are nature-made. Similarly, a building may look different from a car. But these two structures are man-made.

The second problem is *depth*. Specifically, the distance of the object from the camera may influence classification results. For instance, a car may look like a nature-made object from a large distance. These problems motivate us to evaluate new methodologies for segmenting natural and man-made structures.

4.2 Method for segmenting natural and man-made structures

Our method for segmentation of natural and man-made structures is based on the methodology proposed in section 2.2. Here, characteristics "A" and "B" are chosen as "A" = nature-made and "B" = man-made. Therefore, IC filters $\mathbf{a}_i$ are learned from natural scenes, and IC filters $\mathbf{b}_i$ are learned from urban scenes. The block diagram of our method is shown in Figure 4.1.

In the first step, the RGB color bands of the input image are collapsed generating a grayscale image $\mathbf{I}$. Around a pixel $I_{x,y}$ of this grayscale image, let us consider a neighborhood of $2L+1 \times 2L+1$ pixels. This neighborhood is represented by the column vector $\mathbf{n}_{x,y}$. This vector is created by reading the pixels in the neighborhood in a raster scan fashion, i.e., from top to bottom and left to right.

The responses of filters $\mathbf{a}_i$ and $\mathbf{b}_i$ are calculated as

$$\alpha_{x,y,i} = \mathbf{a}_i^{\mathrm{T}} \mathbf{n}_{x,y}, \tag{4.1}$$

$$\beta_{x,y,i} = \mathbf{b}_i^{\mathrm{T}} \mathbf{n}_{x,y}, \tag{4.2}$$

respectively. Let vector

$$\mathbf{u}_{x,y} = [\alpha_{x,y,1}, \alpha_{x,y,2}, \ldots, \alpha_{x,y,p}, \beta_{x,y,1}, \beta_{x,y,2}, \ldots, \beta_{x,y,q}]^{\mathrm{T}} \tag{4.3}$$

contain the all responses $\alpha_{x,y,i}$ and $\alpha_{x,y,i}$. For an simpler notation, let us rewrite $\mathbf{u}_{x,y}$ as

$$\mathbf{u}_{x,y} = [u_{x,y,1}, u_{x,y,2}, \ldots, u_{x,y,k}]^{\mathrm{T}}, \tag{4.4}$$

where $u_{x,y,i} = \alpha_{x,y,i}$ for $i \leq p$, $u_{x,y,i} = \beta_{x,y,i-p}$ for $i > p$, and $k = p + q$. Finally, the kurtosis of $\mathbf{u}_{x,y}$ is computed as

$$K_{x,y} = \frac{\frac{1}{k}\sum_{i=1}^{k}[u_{x,y,i} - \mu_u]^4}{\left\{\frac{1}{k}\sum_{i=1}^{k}[u_{x,y,i} - \mu_u]^2\right\}^2}, \tag{4.5}$$

where μ_u is the mean value of vector $\mathbf{u}_{x,y}$. Since kurtosis of $\mathbf{u}_{x,y}$ is calculated for every possible (x, y), values $K_{x,y}$ form a kurtosis map.

For this application of our methodology, it is noticed that some locations x, y may produce extremely high $K_{x,y}$ values. These singular high values impairs the visualization and segmentation of the entire kurtosis map. Therefore, a simple luminance compression is applied to the kurtosis map. Specifically,

$$K'_{x,y} = \log^*(K_{x,y}), \tag{4.6}$$

where $\log^*(a) = 0$ for $a < 1$ and $\log^*(a) = \log(a)$ for $a \geq 1$.

Now, let's revisit our hypothesis illustrated in Figure 2.4. When image $\mathbf{n}$ exhibits characteristic "A", the response pattern is similar to Figure 2.4(a). When image $\mathbf{n}$ exhibits characteristic "B", the response pattern is similar to Figure 2.4(b). Finally, kurtosis of the response pattern in Figure 2.4(a) is expected to be higher than that of Figure 2.4(b).

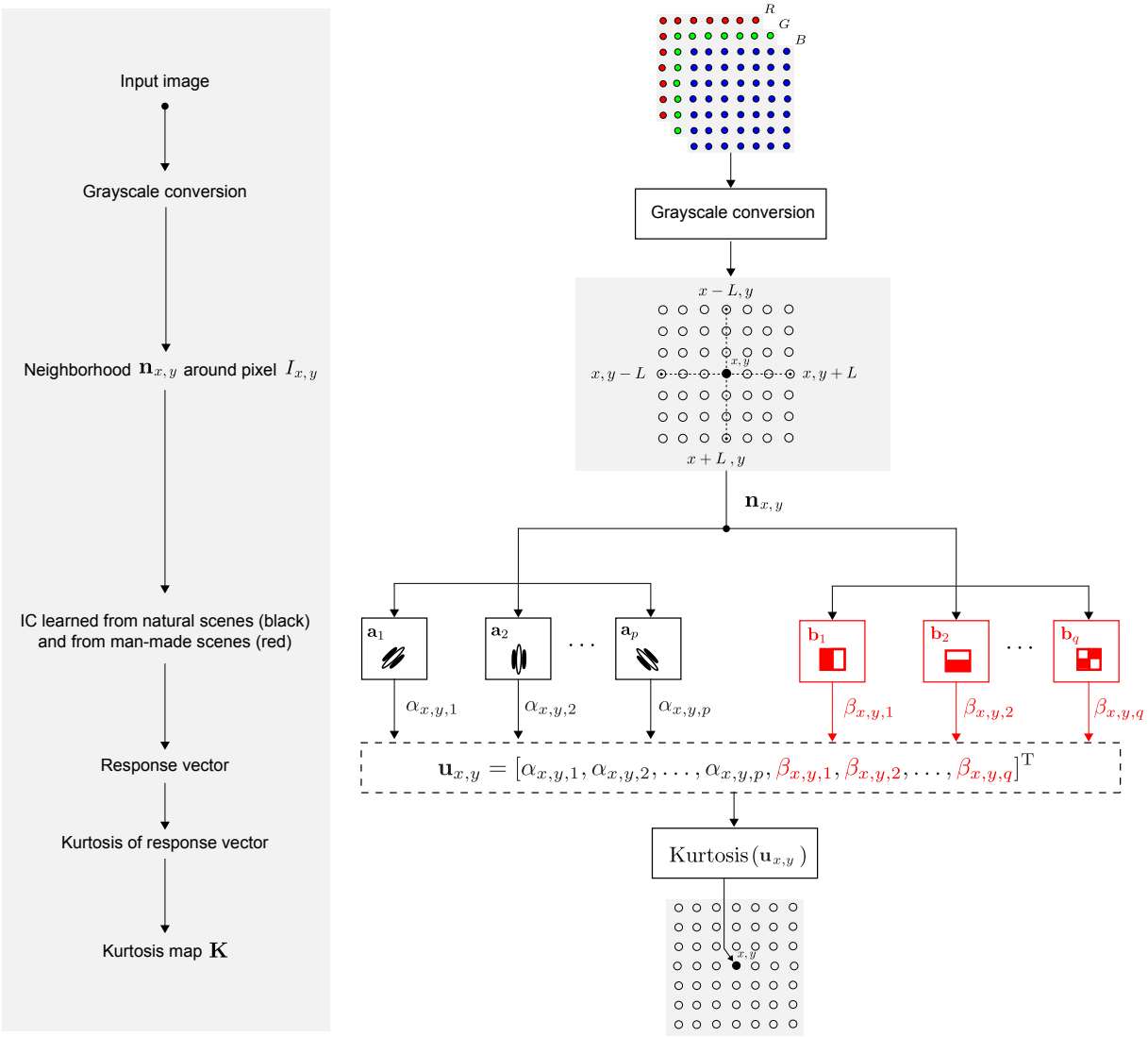

Figure 4.1: **Overview for proposed method for segmentation of natural and man-made structures**. An input image patch is converted into a grayscale image $\mathbf{I}$. Around a pixel $I_{x,y}$ of this grayscale image, consider a neighborhood of pixels represented by $\mathbf{n}_{x,y}$, where (x, y) represents coordinates of a pixel. Two sets of IC filters are represented by $\mathbf{a}_i$ and $\mathbf{b}_i$. Notice that $p \gg q$. Filters $\mathbf{a}_i$ are learned from natural scenes. Filters $\mathbf{b}_i$ are learned from urban scenes, i.e., scenes which contain only man-made objects. The responses of filters $\mathbf{a}_i$ and $\mathbf{b}_i$ are represented by $\alpha_{x,y,i}$ and $\beta_{x,y,i}$, respectively. Vector $\mathbf{u}_{x,y}$ contain both responses $\alpha_{x,y,i}$ and $\beta_{x,y,i}$. The kurtosis map $\mathbf{K}$ is generated by computing the kurtosis of $\mathbf{u}_{x,y}$ for every (x, y). Notice that there are two differences between this block diagram and that for DOF segmentation shown in the previous chapter. The first difference is that there is no log-transformation. The second difference is the choice of filters $\mathbf{b}_i$. Here, they are learned from urban scenes. For DOF segmentation, there is only one filters $\mathbf{b}_i$, which is a very low frequency filter.

Applying our hypothesis for the problem of segmentation of natural and man-made structures, we have that characteristic A = nature-made and B = man-made. Therefore, when input image $\mathbf{n}_{x,y}$ contains a man-made object, kurtosis of $\mathbf{u}_{x,y}$ is expected to be higher than that when $\mathbf{n}_{x,y}$ contains a nature-made object. This is verified in the section *Experiment and results*.

Generating final segmented image

After computing the kurtosis map, it is necessary to generate the final segmented image. This process consists of two steps, i.e., *morphological reconstruction* and *thresholding*. The first process is as follows:

1. **Define**

$$J_{x,y} = \begin{cases} -K_{x,y} & \text{if } x, y \text{ is on the map borders,} \\ M & \text{otherwise,} \end{cases}$$

where $M < -K_{x,y}\ \forall x, y$. Here, it is used $M = \min_{x,y}\{-K_{x,y}\} - 1$.

2. **While** $J_{z+1} \neq J_z$ **do**

$$J_{x,y} \leftarrow \min\{[J \oplus B]_{x,y}, -K_{x,y}\}, \tag{4.7}$$

where $[J \oplus B]_{x,y} = \max_{(x',y') \in N_B}\{J(x - x', y - y')\}$ is the dilation of matrix $\mathbf{J}$ by a flat structuring element $\mathbf{B}$. Here, it is used a 3×3 flat element so that N_B is an 8-connectivity neighborhood.

end of while.

3. **Return** $\mathbf{J}^*$.

After this procedure is executed, the resulting map after filling holes is given by $\mathbf{J}^*$. The *thresholding* step is as follows:

$$A_{x,y} = \begin{cases} 1 & \text{if } J^*_{x,y} > T \\ 0 & \text{otherwise,} \end{cases} \tag{4.8}$$

where $T \in \mathbb{R}$ is a threshold, and $\mathbf{A}$ is the binarized version of $\mathbf{J}^*$. The threshold is set simply as $T = 130$ (after rescaling $J^*_{x,y}$ values in the range $[1, 255]$). Finally, the segmented image is found by multiplying the original image $\mathbf{I}$ by the binary image $\mathbf{A}$.

4.3 Experiment and results

Learned IC filters

This section describes how IC filters $\mathbf{a}_i$ and $\mathbf{b}_i$ are learned. In our model for segmentation of natural and man-made structures, filters $\mathbf{a}_i$ must be learned from natural scenes, and filters $\mathbf{b}_i$ must be learned from man-made scenes. Furthermore, the number of filters $\mathbf{a}_i$ must be much greater than that of filters $\mathbf{b}_i$.

Filters learned from natural scenes were previously presented in Figure 3.2. Therefore, we use that set of 255 IC filters as filters $\mathbf{a}_i$. Notice that the low-frequency filter highlighted in red in Figure 3.2 is not used. If the low-frequency filter is used the system will segment depth of field. And this is not the goal.

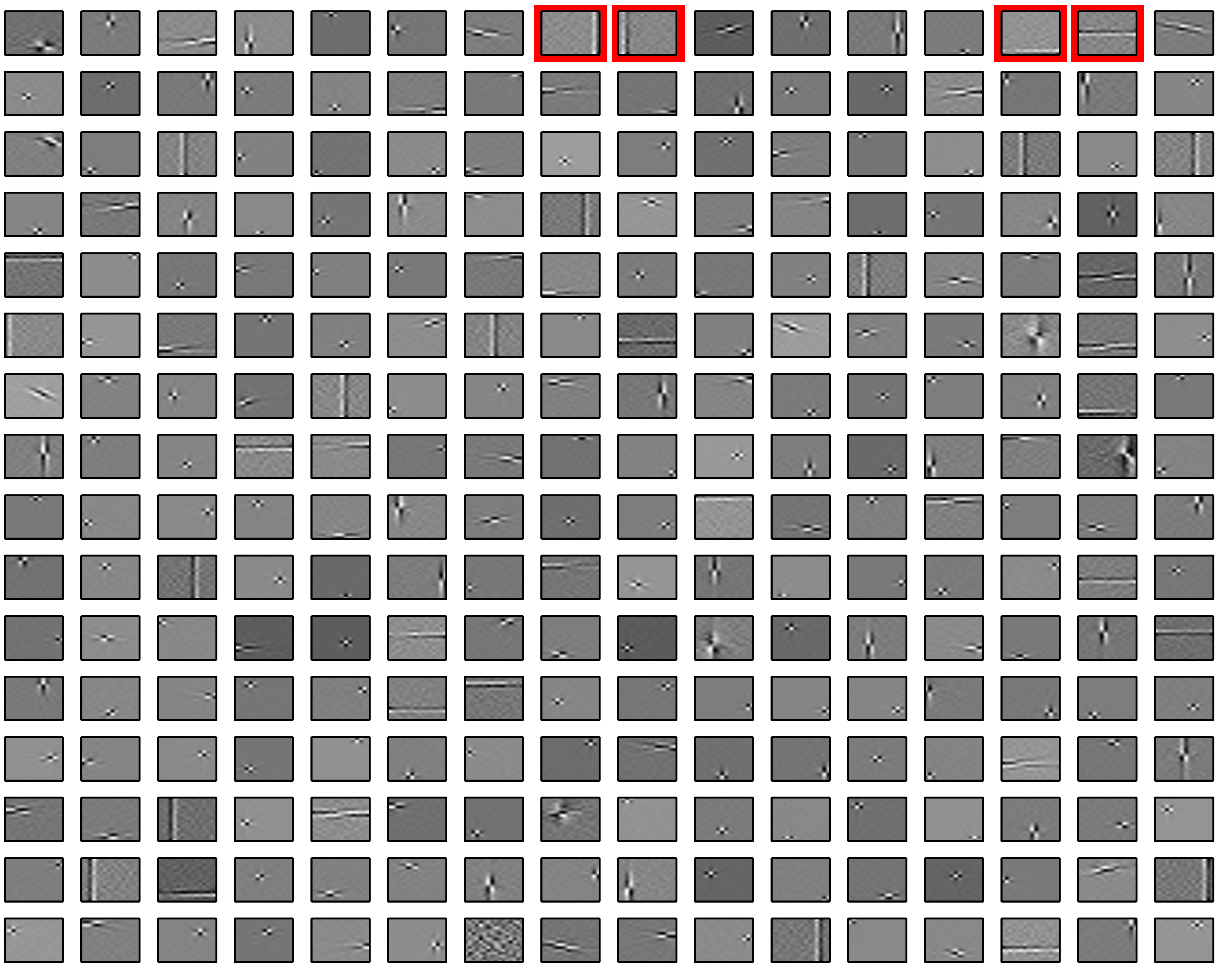

Figure 4.2: **Independent component filters learned from urban scenes**. There are 256 IC filters exhibited in the figure. Each filter is represented by a square of dimensions 16 x 16 pixels. This bank of IC filters has been learned when the input variables were pixels of urban scenes which did not contain nature-made structures. The ICA algorithm used for learning was the FastICA algorithm [69]. There are four IC filters highlighted in red. These four IC filters are used as filters $\mathbf{b}_i$ in our method.

In order to learn filters $\mathbf{b}_i$, urban scenes which do not contain nature-made structures were carefully chosen from different image datasets. From these images, 50,000 patches of 16×16 pixels were extracted in a non-overlapping fashion. This set of image patches was used as input for the FastICA algorithm. The IC filters learned from urban scenes are shown in Figure 4.2.

Since the number of filters $\mathbf{b}_i$ should be very small, only four IC filters in Figure 4.2 were used as filters $\mathbf{b}_i$. These four IC filters are highlighted in red. The reason why these specific filters are chosen is that they are not localized in space like IC filters learned from natural scenes. In this way, there are 255 filters $\mathbf{a}_i$ learned from natural scenes, and four filters $\mathbf{b}_i$ learned from urban scenes.

Notice that the IC filters learned from urban scenes are visually different from those learned from natural scenes. While the IC filters learned from natural scenes are visually similar to Gabor functions, IC filters learned from urban scenes seem similar to Haar functions. Unfortunately, Haar functions are difficult to parametrize. Therefore, quanti-

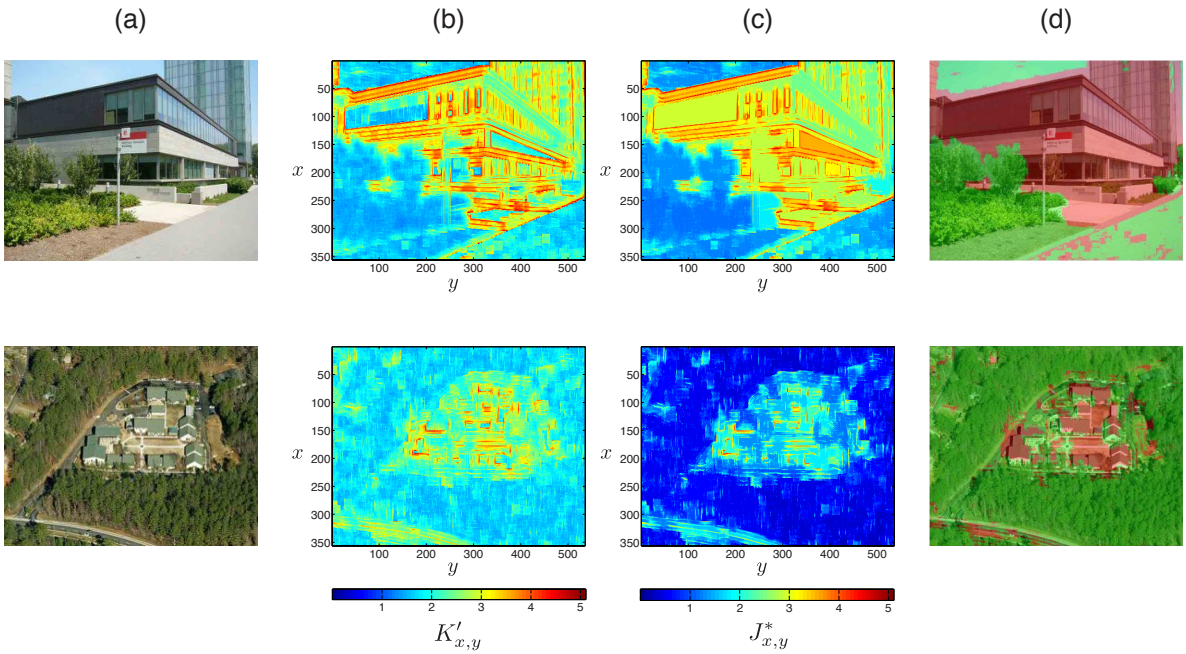

Figure 4.3: **Segmentation of natural and man-made structures by using kurtosis of responses of IC filters**. (a) Original images. (b) Kurtosis maps. Regions containing man-made objects exhibits higher kurtosis $K'_{x,y}$. (c) Kurtosis map after morphological reconstruction. In this map, kurtosis values are represented by $J^*_{x,y}$. (d) Segmented images. Image regions highlighted with the color green indicate nature-made structures. Image regions highlighted with the color red indicate man-made structures.

tative analysis for these IC filters is a theme for future work. It should be noticed that the similarity to Haar function is an interesting fact due to the relation between IC filters and the receptive fields of cells of the primary visual cortex.

Results for segmentation of natural and man-made structures

In order to demonstrate how kurtosis of IC filters represents natural and man-made structures, we have selected an image from the York Urban Line Segment Database [85] and one from the Internet. These images are shown in Figure 4.3(a). Similar to all test images in this work, image size is 356×536 pixels. Furthermore, images are processed in neighborhoods $\mathbf{n}_{x,y}$ of 16×16 pixels.

The kurtosis maps generated by our methodology are shown in Figure 4.3(b). As expected, kurtosis is higher for regions containing man-made objects. Kurtosis maps after morphological reconstruction are shown on Figure 4.3(c). Segmented images obtained from kurtosis maps are shown in Figure 4.3(d).

In section 1.3.2, I described several applications of segmentation of natural and man-made structures. It is important to notice that different applications may imply different conditions of image quality. For example, image quality can be very different comparing a speeding airborne device to a still device on the ground. Therefore, it is important to test image processing methods on images acquired from different real-world platforms.

Here, we test our system on images datasets acquired by still devices and moving devices (the characteristics of these datasets will be discussed later). From the datasets,

Table 4.1: **Performance for segmentation of natural and man-made structures**. These results are computed over 40 images randomly selected from datasets acquired by still and moving devices. The proposed method is compared to the conditional random field (CRF) approach [83], and mean-squared-error (MSE) approach [84].

	Proposed method	CRF [83]	MSE [84]
Average of F-values	0.72	0.30	0.54
Standard deviation of F-values	0.12	0.31	0.19
Average processing time	6 s	3 s	6s

we have selected 40 images (20 images are from still devices, and 20 images from moving devices). For all 40 images, we have manually created ground-truth masks. These masks are binary images in which pixels with values "0" and "1" indicate "nature-made" and "man-made" pixels in the original image, respectively. In this way, ground-truth masks are used to determine which pixels are correctly or wrongly classified in segmented images. For all images, the performance of segmentation methods is quantified using F-measure (see Eq. 3.12). For the calculation of the F-measure, tp is the number of "man-made" pixels correctly classified as "man-made", fp is the number of "nature-made" pixels wrongly classified as "man-made", and fn is the number of "man-made" pixels wrongly classified as "nature-made".

Here, we compare the performance of our method with the conditional random field approach [83] and our previously proposed system based on mean squared error [84]. This specific system is described in Appendix D. The reason why these methods are chosen for comparison is that they are fast and work on any type of images.

Table 4.1 shows the results for the segmentation of natural and man-made structures. In this table, columns indicate different methods. The first row shows the average of F-values over the images in the dataset. The second row shows the standard deviation of F-values. The last row shows the average time required to process an image.

From Table 4.1, it is clear that the proposed measure offer a higher F-measure than the other methods. Furthermore, the average processing time per image required by the proposed method was 6 seconds. For the conditional random field approach, the average processing time was 3 seconds. Finally, the average processing time for the mean-squared approach was 6 seconds. In this way, the conditional random field approach is the fastest method. Specifically, this approach is two times faster than the proposed method. In the following sections, we are going to further investigate the performance of these methods.

Case study: still devices

In order to test our system on images acquired using still devices, two image datasets are used. Specifically, the McGill Calibrated Color Image Database [68] and the York Urban Line Segment Database [86]. Both datasets consist of high quality images acquired using still cameras. The results for samples of these datasets are shown in Figures 4.4 and 4.5.

Let us start discussing segmentation results by the conditional random field approach in Figures 4.4(c) and 4.5 (c). From these images, one can clearly see that the conditional random field approach has lower performance than the other methods for the sampled images. For instance, all image regions in the two last images in Figure 4.5(c) are segmented as nature-made. One can also see that this approach has an interesting characteristic. Specifically, it produces very "smooth" segmented images. The reader should notice that this characteristic can be advantage in some cases.

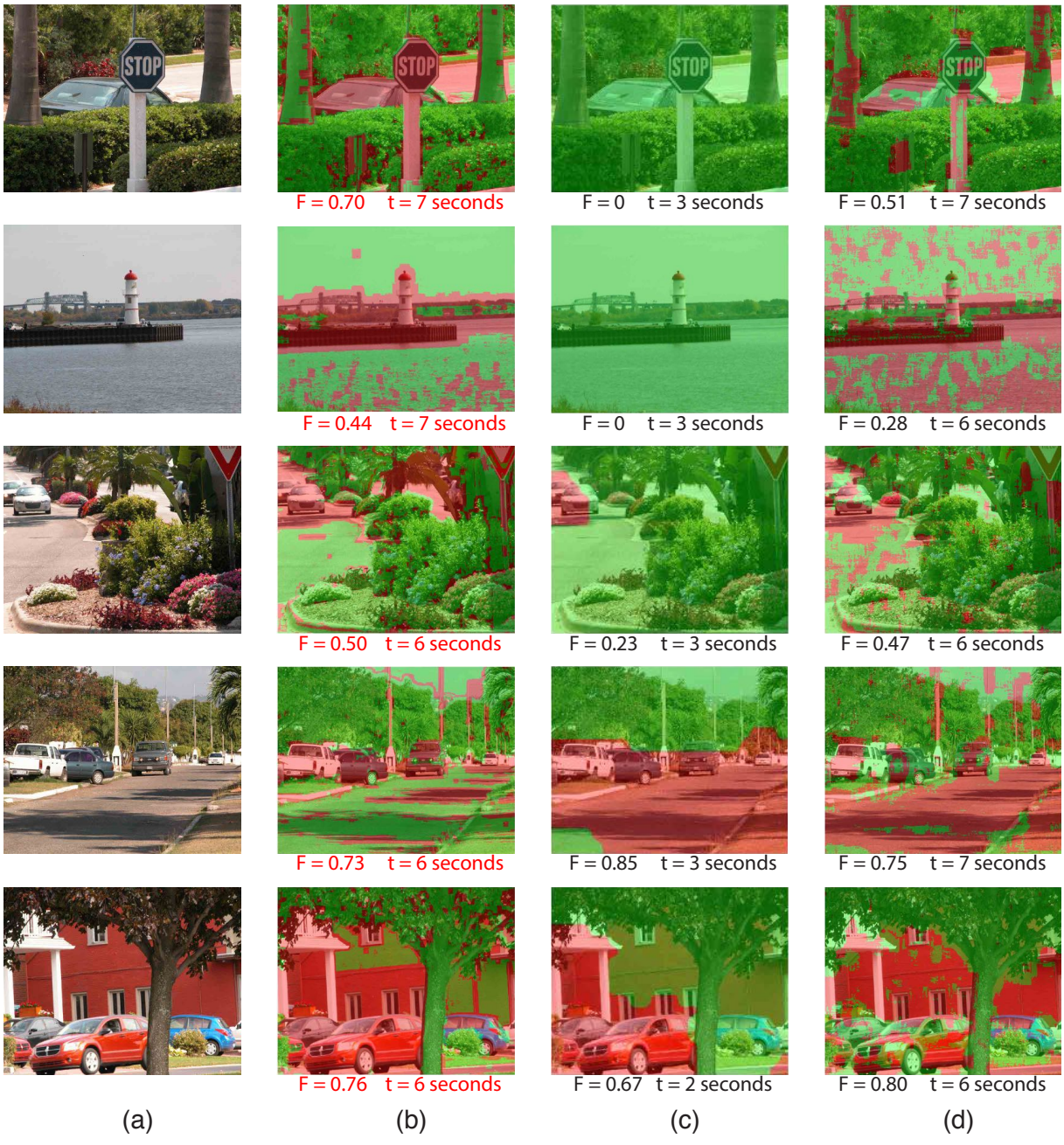

Figure 4.4: **Segmentation of natural and man-made structures for samples of the McGill Calibrated Color Image Database**. (a) Original images. (b) Proposed method. (c) Conditional random field approach [83]. (d) Mean squared error approach [84]. Image regions highlighted with the color green indicate nature-made structures. Image regions highlighted with the color red indicate man-made structures.

Figure 4.5: **Segmentation of natural and man-made structures for samples of the York Urban Line Segment Database**. (a) Original images. (b) Proposed method. (c) Conditional random field approach [83]. (d) Mean squared error approach [84]. Image regions highlighted with the color green indicate nature-made structures. Image regions highlighted with the color red indicate man-made structures.

From Figures 4.4(d) and 4.5(d), it is evident that the mean squared error approach produces noisy segmented images. This will also be noticed for other test images in this work. However, the segmentation result of this method seems more accurate than that of the conditional random field approach.

Finally, let us discuss the result of the proposed method, i.e., Figures 4.4(b) and 4.5(b). For all test images, the proposed method seems to provide more accurate segmentation than that provided by the conditional random field approach. Furthermore, the proposed method clearly provides segmented images less noisier than those provided by the mean squared error approach.

Case study: moving devices

In order to analyze the performances of segmentation methods for images acquired using moving devices, two image datasets were used. The first dataset consists of images acquired by a micro aerial vehicle (MAV) [87]. This specific vehicle is a quadcopter drone flying at low altitude (up to 10-20 meters from the ground), very close to building facades, along a path of two kilometers in the streets of Zurich.

The second dataset consists of images acquired from the well-know Google Stree View service. These images are originated from large 360 degrees panorama scenes acquired by moving cars. Here, the StreetView images were acquired in the same streets of Zurich flow by the quadcopter. These two datasets are part of an matching experiment in which the visual characteristics of buildings are used for autonomous real-time localization of drone devices [87].

Figures 4.6 and 4.7 show the results of segmentation of nature-made and man-made structures for samples from the two datasets. By observing Figure 4.6(a), especially the last scene, one can clear notice the Barrel distortion on images acquired by the quadcopter. In case of Google Street View images in Figure 4.7(a), it is possible to observe by close inspection strong "blocking" artifacts. For confirmation, the reader can zoom-in on the images of the digital version of this thesis. Such artifacts are likely due to the lossy JPEG encoding used to compress these image files.

From Figures 4.6 and 4.7, we judge that the proposed method exhibits a more accurate segmentation than that of the conditional random field approach and that of the mean squared error approach. For instance, the proposed method fairly detects vegetation while buildings and cars are mainly segmented as man-made. This is true for the micro aerial vehicle image dataset and the Google Street View image dataset. In regard of the latter, however, the proposed method often segments sky regions as man-made. The reason for this failure is the presence of "blocking" artifacts which creates luminance edges in the sky. However, the proposed method still exhibits a more accurate performance than that of other methods.

It is important to discuss the above results in regard of the applications of segmentation of natural and man-made structures cited in the introductory chapter. In this regard, let us discuss the choice of datasets used in the analysis in this thesis. The datasets included images acquired from both still and moving devices. Strategically, moving devices included both ground and aerial vehicles. Therefore, the performance of methods for segmentation of natural and man-made structures reported in this thesis is likely to be reproduced or consistent under different situations or applications.

Let us consider the use of vegetation detection to support autonomous driving in complex environments. Firstly, it is important to notice that autonomous driving in the

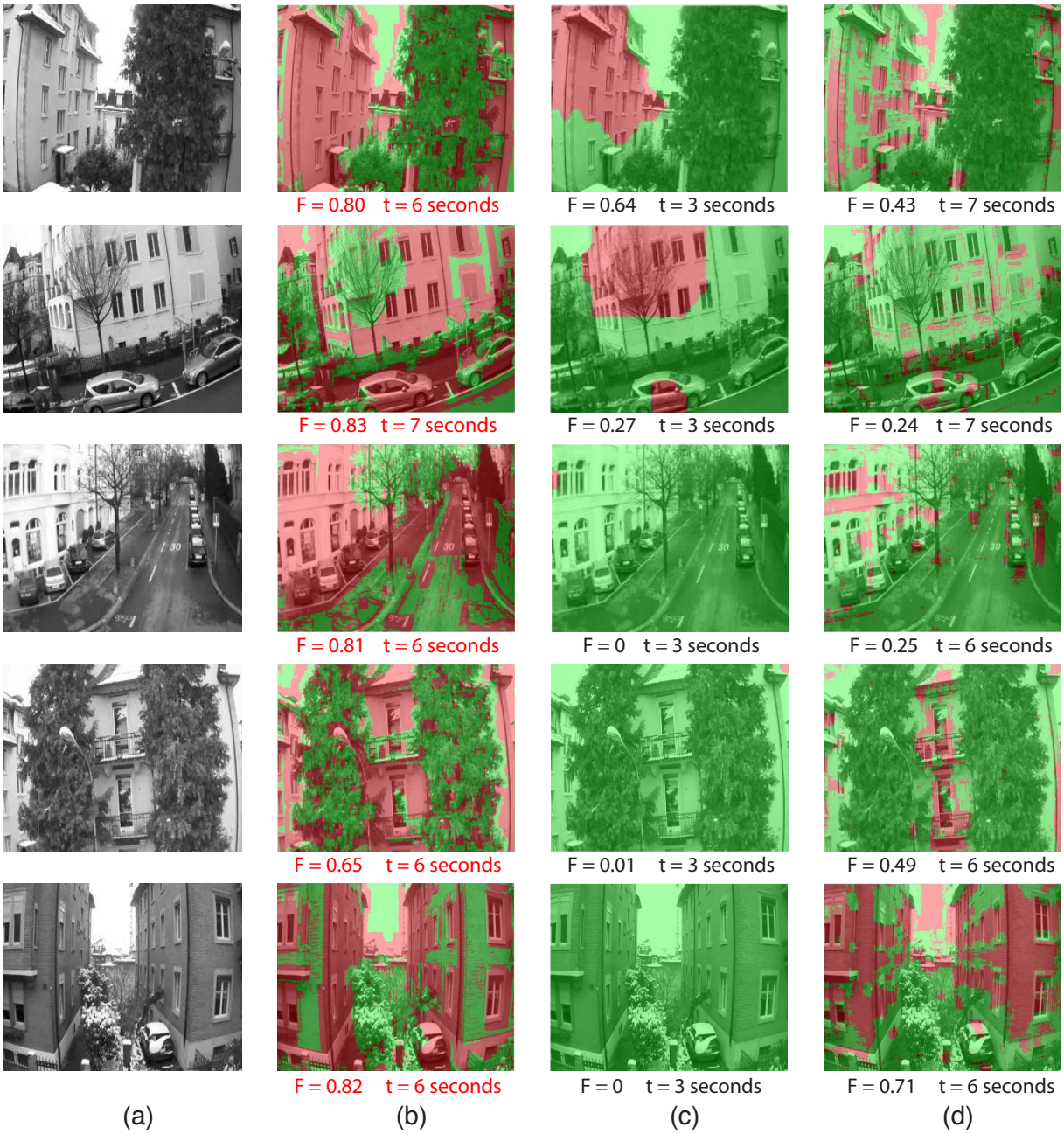

Figure 4.6: **Segmentation of natural and man-made structures for samples of the micro aerial vehicle image dataset**. (a) Original images. These images (especially the last image) exhibit Barrel distortion. (b) Proposed method. (c) Conditional random field approach [83]. (d) Mean squared error approach [84]. Image regions highlighted with the color green indicate nature-made structures. Image regions highlighted with the color red indicate man-made structures.

Figure 4.7: **Segmentation of natural and man-made structures for samples of the Google Street View image dataset**. (a) Original images. These images exhibit "blocking" artifacts. (b) Proposed method. (c) Conditional random field approach [83]. (d) Mean squared error approach [84]. Image regions highlighted with the color green indicate nature-made structures. Image regions highlighted with the color red indicate man-made structures.

real-world is still initial, lacking proper regulation and standards of performance and safety. Furthermore, vegetation detection is only one feature of a much more complex intelligent system. Thus, it is not straightforward to define what performance level of vegetation detection is appropriate or required to support autonomous driving.

Still, it has been shown that proper vegetation detection improves the "overall ability to autonomously avoid rigid obstacles without being overly afraid of bushes or tall grasses" [21]. The reason is because sensors used in autonomous vehicles often mistake bushes as false obstacles [21]. Thus, it is desirable that the performance of the system used for vegetation detection be as high as possible. Taking into consideration the issues related to the use of autonomous driving in the real-world, it is reasonable to assume that the performance reported here and elsewhere is still far from the necessary for safe and efficient execution. For instance, the system proposed in [21] still wrongly classifies image regions containing ground as vegetation, 21% of the time. This error rate may be easily considered unacceptable for an application which directly involves the safety of human lives.

Shortcomings of the proposed method

We provide here an analysis of shortcomings of the proposed method. By understanding these shortcomings, one can easily predict when the method will work or not. The first shortcoming is the poor segmentation of object boundaries. Specifically, our method (wrongly) segments pixels around object boundaries. For example, in the first image of Figure 4.5(b), boundaries of buildings are not precisely segmented. Notice that this is the same shortcoming previously presented for the segmentation of depth-of-field images. Therefore, one must expect such type of failure.

For this application, the proposed method also exhibits another shortcoming related to boundary processing. Specifically, our method wrongly segments boundaries of nature-made objects which stand against a background formed by sky regions. In this situation, the entire image should segmented as natural. However, our method wrongly segments the pixels around the boundaries of the natural object as man-made. This can be observed in the last image of Figure 4.5(b). Notice that boundaries of the tree are segmented as man-made.

The third shortcoming of the proposed method is the processing of roads. Specifically, if road areas do not exhibit any texture, like sky regions, they will be segmented as nature-made structures. This can be observed in some of the presented test images.

4.4 Conclusion

This chapter has presented the method for segmenting natural and man-made structures based on our methodology for visual scene analysis. Here, the required filters $\mathbf{a}_i$ and $\mathbf{b}_i$ consist of IC filters learned from nature scenes and urban scenes, respectively. Since filters $\mathbf{b}_i$ are learned from urban scenes, the kurtosis of filter responses is high for input images containing man-made structures, and it is low for input image containing nature-made structures.

Due to the different image conditions on real-world applications, our method was tested on two types of datasets. The first and the second dataset consist of images acquired by still and moving devices, respectively. In comparison to other analyzed

systems, our method provides a more precise segmentation, especially in case of the image datasets acquired by moving devices. This is an important advantage which makes the proposed method attractive for real-world applications.

In regard of performance, the shortcomings of our method were also presented. The most serious shortcoming is related to the segmentation of road regions. As it is, the system may wrongly classify road regions as nature-made. In this regard, discriminating road regions from sky regions is a difficult task. This task likely requires the use of color information or some type of high-order processing architecture. Future works will focus on this issue.

Taking these considerations into account, we conclude that the proposed method is a promising and interesting approach for the segmentation of nature-made and man-made structures due to its performance and low computational complexity.

Chapter 5

Measuring the perception of complexity in streetscapes

5.1 Introduction

What makes us perceive or decide that a visual scene"A" is more complex than a scene "B"? Attneave showed that for scenes containing abstract shapes, certain visual characteristics (which he named symmetry, curvedness, angular variation, etc) was related to the perception of visual complexity [88]. By combining these characteristics into a single equation, Attneave created an objective measure which was correlated with human judgments on visual complexity.

The characteristics of spatial frequency have also been shown to influence the perception of visual complexity. Specifically, it is reported that the amplitude of high-frequency components must be preserved for complex objects to be recognized [89–91]. Similarly, specific relationships among frequency components in the phase spectrum are crucial for visual recognition of complex scenes [92]. These results have been extended by many other studies in vision research, involving many types of visual scenes.

Based on the characteristics of spatial-frequency, Näsänen et al. derived a complexity measure defined as the product between the effective image area and median frequency of the Fourier spectrum [93]. Chikhman et al. used the components of this measure to analyze complexity in ancient Egyptian writing and contour images [94]. Notice that Näsänen's method can be applied to real-world scenes.

It has also been shown that the presence of image edges is related to visual complexity [95]. This inspired a simple and efficient measure known as *perimeter detection*. The measurement consists of counting the number of pixels which form image edges. This procedure can be easily applied to real-world scenes by using edge-detection algorithms.

In order to measure visual clutter, a concept closely related to complexity, Rosenholtz et al. proposed a framework called feature congestion. Within this framework, several image characteristics such as contrast, color and orientation are combined into a vector space [96]. Clutter is then determined by the covariance of the space calculated at each location of the image.

Another line of research was based on the idea of computing visual complexity according to the definitions of information theory [97]. In this view, a visual scene is considered an information source, and its visual complexity is determined by the amount of information associated to its statistical distribution.

An example of information based measure is the size in bytes of the image digital file created according to coding standards such as JPEG and GIF. Theoretically, file size

should increase as the amount of information increases. The JPEG file size has been used in many perception works due to its high correlation with subjective judgments of complexity. Forsythe et al. provides an extensive analysis of the performance of JPEG and also of perimeter detection [98].

Another example of information based measure is the subband entropy [39]. The subband entropy is defined as the Shannon entropy of wavelet coefficients used to encode an image.

Other methods have also been considered to evaluate visual complexity in urban environments. For instance, Elsheshtawy used a manual approach to segment meaningful elements of street houses such as windows, doorways and overall volumes of facades [99]. Complexity was then measured based on the number and variety of those elements. Cooper also used a manual technique to segment street skylines, i.e., edges formed between the boundaries of buildings and the sky [100]. Then, he used fractal dimension to assess the complexity of these skylines.

Applications

The introductory chapter describes the effect of streetscape complexity on the behavior of drivers and pedestrians. For instance, streetscape complexity influences the speed control [43], reaction time and the perception of hazard of car drivers. In regard of pedestrians, streetscape complexity influences the visual interest evoked by streets. In this way, there several possible applications of measuring the perception of complexity in streetscapes.

For engineering, a measure of streetscape complexity could be used to create or aid human inspired speed control mechanisms. Furthermore, it could be used to evaluate the perceptual load and reaction time of drivers for each city streets. For urban planning, a measure of streetscape complexity could be used to evaluate visual interest of city streets. Also, it could be used to determine routes which maximize visual interest of pedestrians. However, although the effects of streetscape complexity on the behavior of drivers and pedestrians is well-known, the above applications have not yet been evaluated in scientific literature.

Open problems

The main problem of many works on visual perception of the environment is handling nighttime stimuli. For example, in our previous work, we have analyzed the complexity in streetscape images by using the statistics of local contrast [101]. We have found that these statistics are highly correlated with subjective judgments for daytime images. However, being similar to conventional measures of complexity, they produce poor results when nighttime images are considered.

Since city streetscapes are experienced or appreciated throughout the day, proper evaluation for nighttime scenery is just as important as for those in daytime. In [102], we introduce a new measure of visual complexity which exhibits a high and robust performance over different time scenarios. This measure is formed by combining the statistics of local contrast with those of local spatial frequency. This chapter describes such measure.

5.2 Proposed measure of streetscape complexity

In order to measure the perception of complexity, our new method uses the statistics of local spatial frequency and local contrast of the streetscape image. The block diagram is shown in Figure 5.1.

In the proposed method, the RGB color bands of the streetscape image are collapsed generating a grayscale image $\mathbf{I}$. Around a pixel $I_{x,y}$ of this image, let us consider a neighborhood of $2L+1 \times 2L+1$ pixels. This neighborhood is represented by the column vector $\mathbf{n}_{x,y}$.

The rest of the computer method consists of two workflows. The first workflow (left-hand side of the block diagram in Figure 5.1) calculates the contrast map $\mathbf{C}$ of the input streetscape image. The contrast map is used to represent the values of local contrast. Each value $C_{x,y}$ of this map is calculated as the standard deviation of vector $\mathbf{n}_{x,y}$, i.e.,

$$C_{x,y} = \sqrt{\frac{1}{(2L+1)^2} \sum_{i=1}^{(2L+1)^2} (n_{x,y,i} - \mu_n)^2}, \tag{5.1}$$

where $n_{x,y,i}$ and μ_n represent the i-th element and the mean value of vector $\mathbf{n}_{x,y}$, respectively. The above measure $C_{x,y}$ is also called the RMS contrast. Notice that although RMS contrast is used, there are other definitions or measures of image contrast [103].

The second workflow (right-hand side of the block diagram in Figure 5.1) calculates the kurtosis map $\mathbf{K}$ of the input streetscape. The kurtosis map is used to represent the values of local spatial frequency. This kurtosis map is computed with the same IC filters $\mathbf{a}_i$ and $\mathbf{b}_i$ as used for segmenting depth-of-field in Chapter 3. Here, however, the response $\beta_{x,y,i}$ is as constant c for all x, y (we find that this constraint yields slightly better results).

In Figure 5.1, the proposed measure of complexity is represented by γ, and it is calculated based on the statistics of the maps $\mathbf{K}$ and $\mathbf{C}$. The first statistic is the mean value of $\mathbf{C}$. This mean is represented by μ_C, and computed as

$$\mu_C = \frac{1}{M^2} \sum_{x=1}^{M} \sum_{y=1}^{M} C_{x,y}, \tag{5.2}$$

where $M \times M$ is the size of $\mathbf{C}$.

The second statistic is the standard deviation value of $\mathbf{C}$. This statistic is represented by σ_C, and calculated as

$$\sigma_C = \sqrt{\frac{1}{M^2} \sum_{x=1}^{M} \sum_{y=1}^{M} (C_{x,y} - \mu_C)^2}. \tag{5.3}$$

The third statistic is the skewness of $\mathbf{K}$. This statistic is represented by $skew_K$, and defined as

$$skew_K = \frac{\frac{1}{M^2} \sum_{x=1}^{M} \sum_{y=1}^{M} (K_{x,y} - \mu_K)^3}{\left\{ \sqrt{\frac{1}{M^2} \sum_{x=1}^{M} \sum_{y=1}^{M} (K_{x,y} - \mu_K)^2} \right\}^3}, \tag{5.4}$$

where $M \times M$ and μ_K are the size and the mean value of $\mathbf{K}$, respectively.

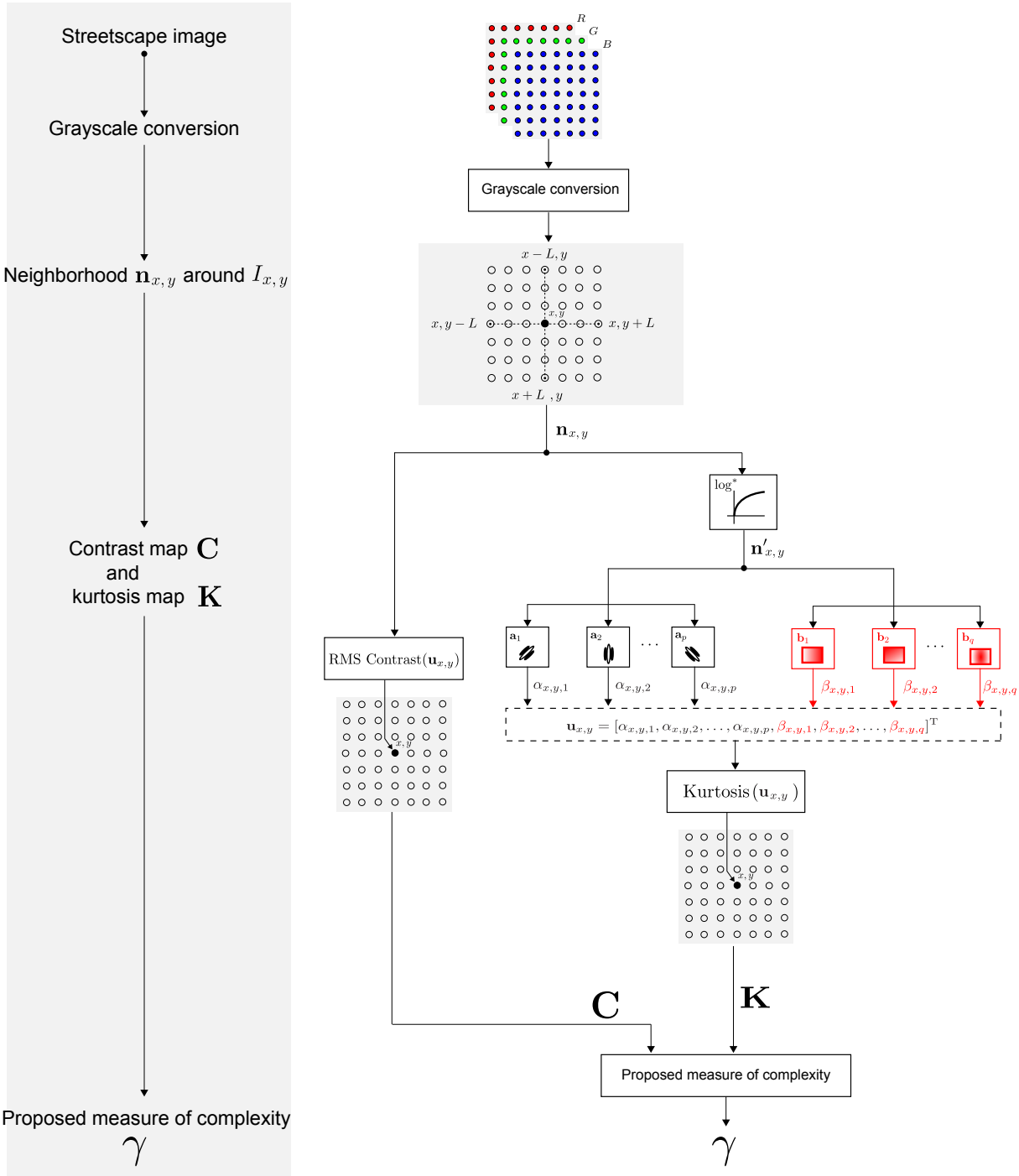

Figure 5.1: **Overview for proposed method for measuring the perception of complexity in streetscapes**. The first workflow (left-hand side of the block diagram calculates the contrast map $\mathbf{C}$ of the input streetscape image. The second workflow calculates the kurtosis map $\mathbf{K}$. The proposed measure of complexity γ is computed based on the statistics of maps $\mathbf{C}$ and $\mathbf{K}$. Notice that the kurtosis map $\mathbf{K}$ is calculated as for DOF segmentation. Therefore, the main difference between this block diagram and that for DOF segmentation is the presence of the contrast map $\mathbf{C}$.

The fourth and last statistic is the kurtosis of $\mathbf{K}$. This statistics is represented by $kurt_K$, and calculated as

$$kurt_K = \frac{\frac{1}{M^2}\sum_{x=1}^{M}\sum_{y=1}^{M}(K_{x,y}-\mu_K)^4}{\left\{\frac{1}{M^2}\sum_{x=1}^{M}\sum_{y=1}^{M}(K_{x,y}-\mu_K)^2\right\}^2}. \tag{5.5}$$

Finally, the proposed measure of complexity γ is given by

$$\gamma = \frac{\mu_C \cdot \sigma_C \cdot skew_K}{kurt_K}. \tag{5.6}$$

The following sections describe how the statistics of $\mathbf{C}$ and $\mathbf{K}$ (i.e., μ_C, σ_C, $skew_K$ and $kurt_K$) relates to the visual structure of streetscapes. The next section shows that measure γ increases with presence of both high-contrast image regions and low-frequency image regions. These type of image regions are shown to drive physiological neuron responses, human attention and emotional event processing [104,105]. This will be further discussed later.

Finally, notice that in Eq. 5.6, we use multiplication of parameters instead of addition. The reason is that these parameters produce different ranges of amplitude. In case addition was used, a parameter with large amplitude could completely "mask" the value of small amplitude parameter. For instance, if $a, b \in \mathbb{R}$, $a + b \approx a$ for $a \gg b$.

5.3 Experiment and results

Streetscapes data

The streetscape dataset consists of 74 scenes. One half of the images were acquired in Al-Kantara and Batna cities in Algeria. The other half was acquired in the cities of Kyoto and Tokyo in Japan. Within the dataset, 40 images were acquired in daytime and 34 images in nighttime.

Images were shot using the camera model Nikon D300S with lens system Nikkor AF-S DX 35mm f/1.8G. The camera was fixed in a tripod in order to avoid artifacts due to camera shaking. Aperture and shutter speed were determined manually according to the lighting conditions in each of the 74 scenes. Image files were recorded in uncompressed color NEF format (Nikon's raw file designation). The size of the raw images was 4288 x 2848 pixels and image quality was 14 bits/pixel.

Figure 5.2 shows some examples of streetscapes from each city. Figure 5.2(a) and 5.2(b) show streetscapes in the Algerian cities of Al-Kantara and Batna, respectively. Figure 5.2(c) and 5.2(d) exhibit Japanese streetscapes in Kyoto and Tokyo cities.

Subjective ranking

For the subjective experiments, images were presented to participants in a 30" display (model Dell UltraSharp 3008WFP). This display's highest resolution is 2560 x 1600 pixels, which prevents images being exhibited in raw size. Therefore, images were pre-processed by *decimation*. This process consists of *low-pass filtering* and then *downsampling* the

Figure 5.2: **Streetscapes.** (a) Al-Kantara. (b) Batna. (c) Kyoto. (d) Tokyo.

image. Low-pass filtering before downsampling is performed so as to avoid *aliasing* artifacts. Here, we used a zero-phase eighth-order low-pass Chebyshev Type I filter with normalized cutoff frequency of 0.8/2. The images were then down sampled by a factor of 2. In this way, the size of the pre-processed images was 2144 x 1424 pixels which can be exhibited on the used display. Finally, decimated images were converted to 8 bit integer arrays so that their pixel's luminance is within the range [0, 255].

Streetscape images were analyzed by 40 participants. Among the participants, 27 were of Japanese nationality, 13 of Algerian nationality, 25 were males, and 15 were females.

The subjects seated at a distance of approximately 80 cm from the display. Each image therefore subtended 37 x 25.12 degrees of visual angle. The maximum spatial frequency in an image was approximately 28.9 cycles/degree horizontally, and 28.3 cycles/degree at vertical orientation.

In order to make the subjective evaluation faster, the participants were initially asked to cluster the streetscapes into three groups: *simple*, *ordinary* and *complex*. In this regard, they were instructed to use their own perception or definition of complexity. Finally, the subjects were asked to sort images inside each group in increasing order of complexity.

After receiving the 74 ranked images from one participant, it was necessary to represent the divisions between *simple* and *ordinary* groups, and between *ordinary* and *complex* groups. These divisions were represented by including two additional rank positions. For example, if the group *simple* contained ten streetscapes, the division between *simple* and *ordinary* groups would occupy the 11th position in the rank. The images in the *ordinary* group would then start from position 12th. In similar manner, another additional position would be considered for the division between *ordinary* and *complex* groups. In this way, the complexity rank returned from one participant has 76 positions, which includes the 74 images plus the two group divisions. It is important to notice that images and group divisions are sorted differently by each of the 40 subjects. Thus, the rank position of a streetscape (or group division) is a random variable. The probability distribution of this variable is computed by counting the number of times v_i in which the image was located by the subjects at each rank position i. This probability distribution

is represented in Figure 5.3.

For each streetscape, the mean r of the probability distribution of rank position is computed by using the standard definition of mean, i.e.,

$$r = \sum_{i=1}^{76}(i \cdot p_i) = \sum_{i=1}^{76}\left(i \cdot \frac{v_i}{40}\right). \tag{5.7}$$

Streetscapes are then finally sorted according to their mean r. Group divisions are also included in the sorting since they also have probability distributions for rank positions.

The plot in Figure 5.4 shows this rank. The vertical and horizontal axis give the mean r and the resulting rank position for each streetscape, respectively. The blue shade in the plot represents the standard deviation of the distributions for the streetscapes. Group divisions are also included, dividing the plot into three areas, *simple*, *ordinary* and *complex*.

It is possible to see that streetscapes in the group *ordinary* have higher standard deviation of rank position than *simple* and *complex* streetscapes. Interestingly, group divisions exhibit lower standard deviations than streetscapes.

The group *simple* consists of 12 scenes: all Algerian streetscapes; six dayscapes and six nightscapes. The category *ordinary* includes 47 scenes: 24 Algerian streetscapes and 23 Japanese streetscapes; 24 dayscapes and 23 nightscapes. The group of complex streetscape is formed by 15 images: two Algerian streetscapes and 13 Japanese streetscapes; 11 dayscapes and four nightscapes.

Algerian scenes dominate the group of simple streetscapes and the lower region of the group *ordinary*. Japanese scenes dominate the higher region of the group of ordinary streetscapes and they correspond to the great majority in the group *complex*.

In groups *simple* and *ordinary*, dayscapes and nightscapes are evenly distributed. However, dayscapes dominate the group of complex streetscapes.

Contrast and kurtosis maps

This section helps the reader understand how RMS contrast and kurtosis maps can be used to represent contrast and spatial frequency in a streetscape. In Fig. 5.5(a), the upper plot shows an array of image edges. Each individual edge is a matrix of 16 x 16 pixels which contains only two luminance intensity values. Specifically, the upper half of each edge is formed by an intensity value higher than that of its lower half. The number δ below each edge is the difference between upper and lower intensity values. From left to right in the array, the luminance difference δ increases.

Since each image edge is a matrix of pixels, we can calculate its RMS contrast by using Eq. 5.1. Specifically, we consider each individual edge as a neighborhood $\mathbf{n}_{x,y}$, and then apply Eq. 5.1. The respective RMS contrast values are given in the colored array of numbers in Figure 5.5(a). Colors are used to highlight low, medium and high values.

Figure 5.5(b) shows how kurtosis map values change. The array of control images is composed of pure two-dimensional cosine gratings of 16 x 16 pixels. In these gratings, horizontal and vertical components of the spatial frequency are constrained to have the same value. This frequency is represented by the number f below each grating. From left to right, the frequency f increases.

Notice that the frequency f is given in cycles/pixel (cpp) instead of cycles/degree. The reason is that frequency segmentation based on filter activity does not take into

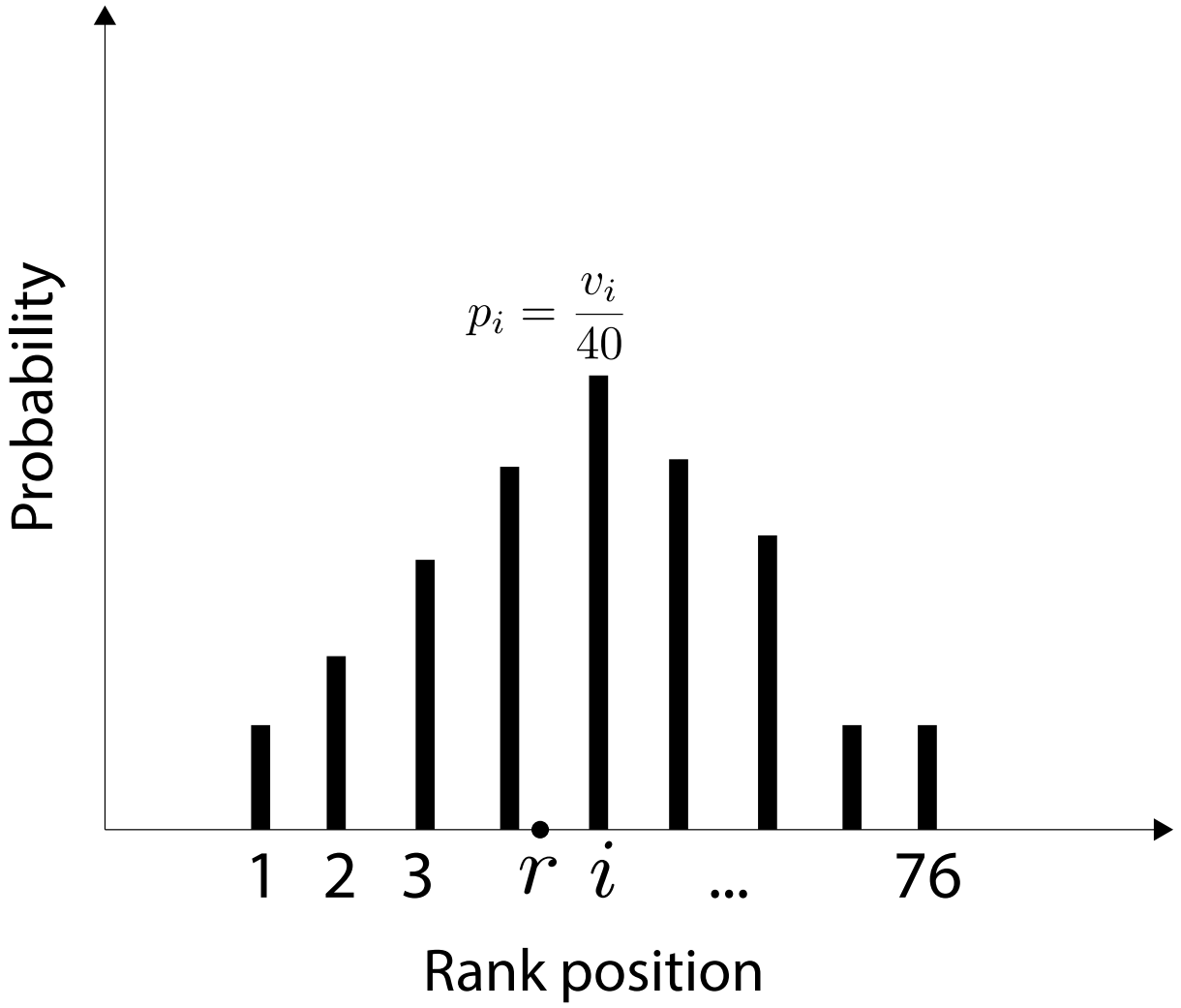

Figure 5.3: **Probability distribution of rank position for one streetscape.** This probability distribution describes how one specific streetscape was ranked by the participants. The v_i is the number of times the image was located at position i by the subjects. Considering 40 subjects, the probability of the image to be ranked at any specific position i is $p_i = \frac{v_i}{40}$. The point r represents the mean of the distribution. Notice that there are 76 possible positions due to the two additional positions for group divisions.

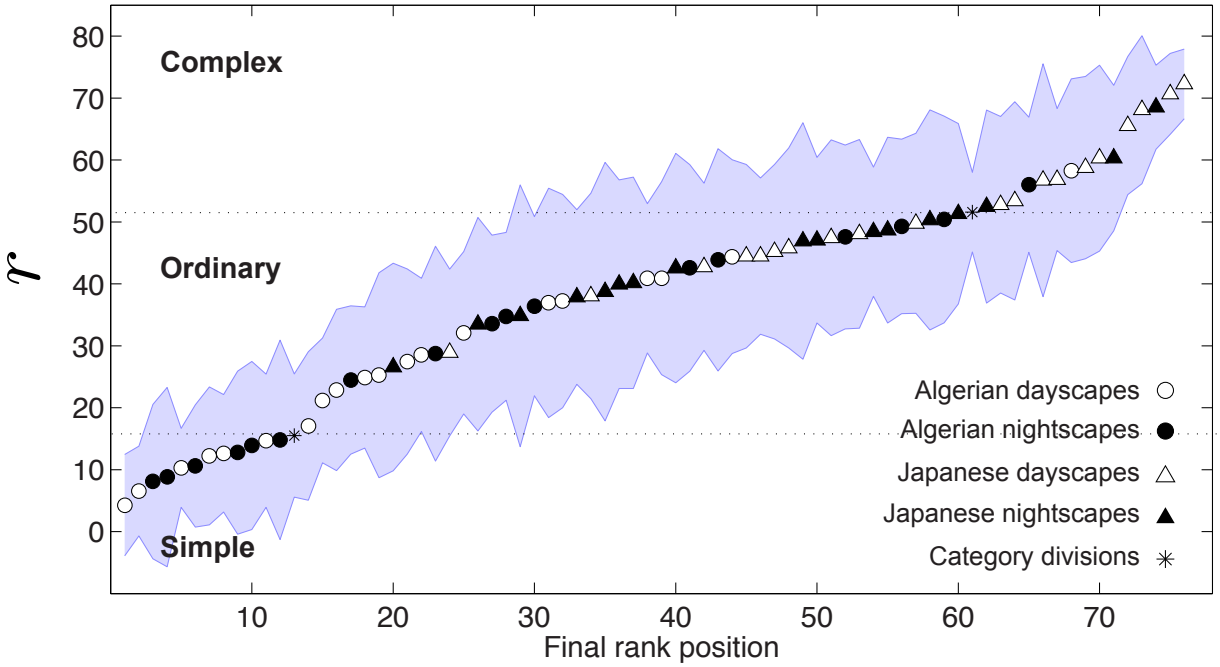

Figure 5.4: **Subjective rank analysis.** Streetscapes are organized in increasing order of r-values. Circles represent Algerian streetscapes. Triangles represent Japanese streetscapes. Unfilled circles/triangles denote dayscapes. Filled circles/triangles denote nightscapes. Stars $*$ represent group divisions. The blue shade represent the standard deviation around r.

account viewing distance. In other words, the process is influenced only by the number of cycles per pixel and not by the number of cycles per degree.

The colored array $K(f)$ contains the respective kurtosis map values calculated when considering each grating one neighborhood. Notice that low-frequency gratings generates high kurtosis which indicates a reduced response activity from the IC filters. High-frequency gratings, however, generate low kurtosis values indicating a dense filter response activity.

In Figure 5.6, true contrast and kurtosis maps are exhibited for an example of streetscape image. These maps were calculated using neighborhoods of 16 x 16 pixels. In the streetscape, objects with luminance which differ from their surroundings generate high values in the contrast map. One can notice, however, that most of the structures present in the scene do not generate such high values of contrast.

In the kurtosis map, low-frequency areas such as the road generate high kurtosis values. On the other hand, textured regions such as the vegetation and the sidewalk have higher energy in high-frequencies generating lower kurtosis values.

Statistics of contrast and kurtosis maps

Figure 5.7 shows the histograms of the contrast and kurtosis maps exhibited previously in Figures 5.6(b) and (c). By using these histograms, one can analyze more precisely the distribution of local contrast and spatial frequency within the streetscape in Figure 5.6(a).

For instance, in Figure 5.7(a), the histogram of the contrast map shows more clearly the number of low-contrast locations in relation to that of high-contrast. However, while

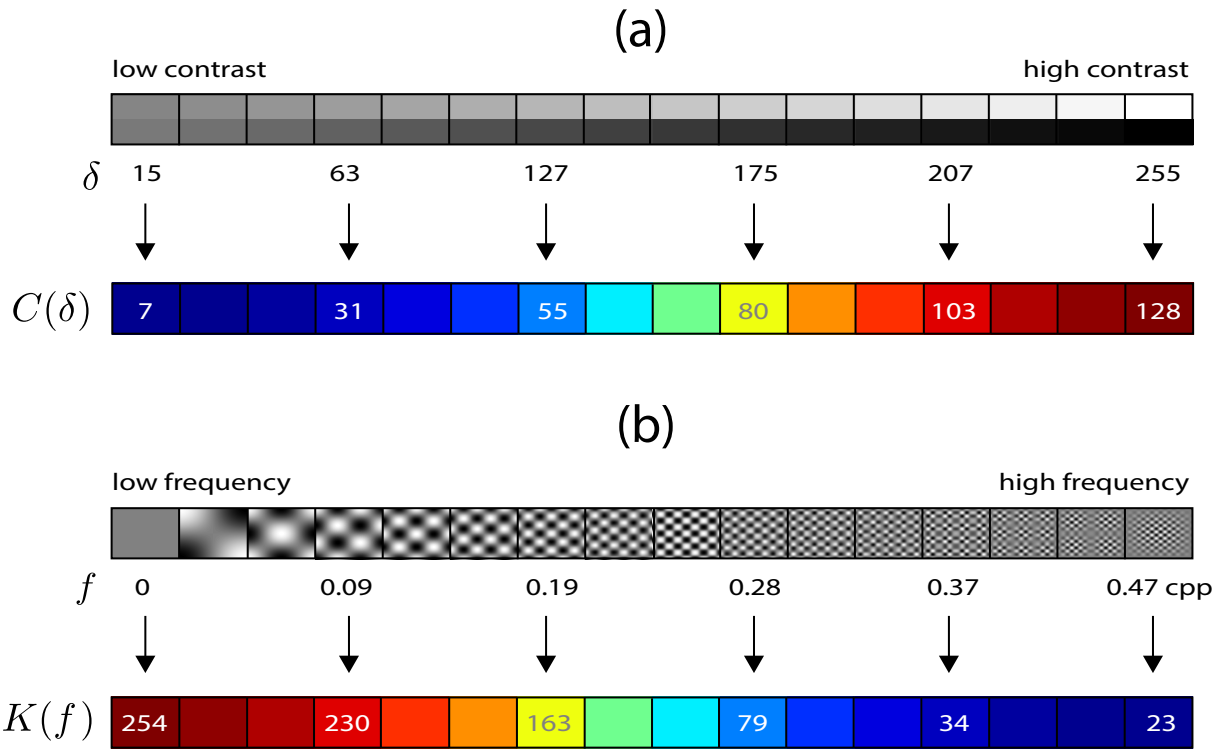

Figure 5.5: **Representation of contrast and spatial frequency content by using RMS contrast and kurtosis.** (a) The plot shows an array of image edges, each of 16 x 16 pixels. The number δ below each edge represents the luminance difference between upper and lower parts. From left to right, this luminance difference increases. Since each edge is a matrix of pixels, we can calculate its RMS contrast by using Eq. 5.1. The colored array of numbers $C(\delta)$ contains the respective RMS contrast values. (b) An array of two-dimensional cosine gratings of 16 x 16 pixels. The number f represents the spatial frequency of each grating. The colored array of numbers $K(f)$ shows the respective kurtosis value generated by the proposed system.

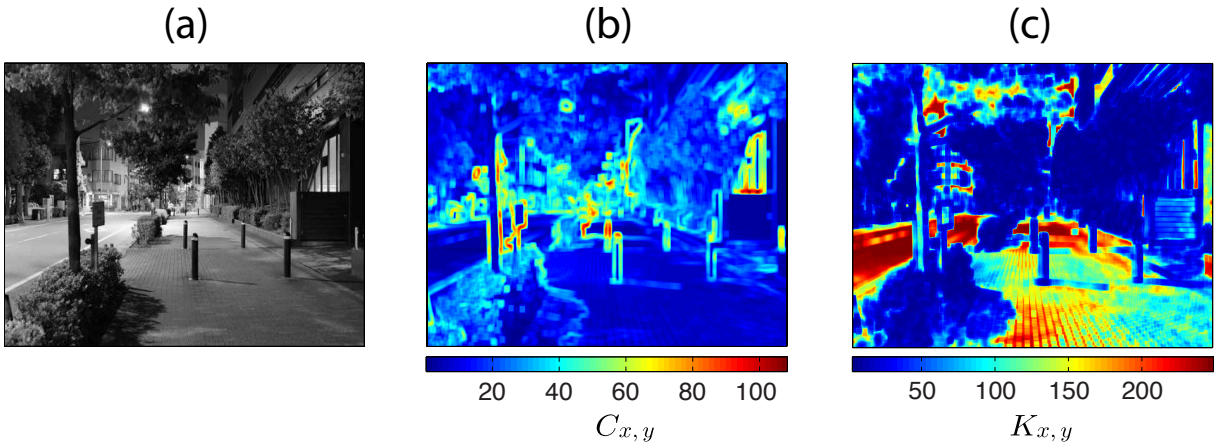

Figure 5.6: **Contrast and kurtosis maps.** (a) Original image. (b) Respective maps **C** and (c) **K**. Colormaps associated with RMS contrast and kurtosis values are shown below each map.

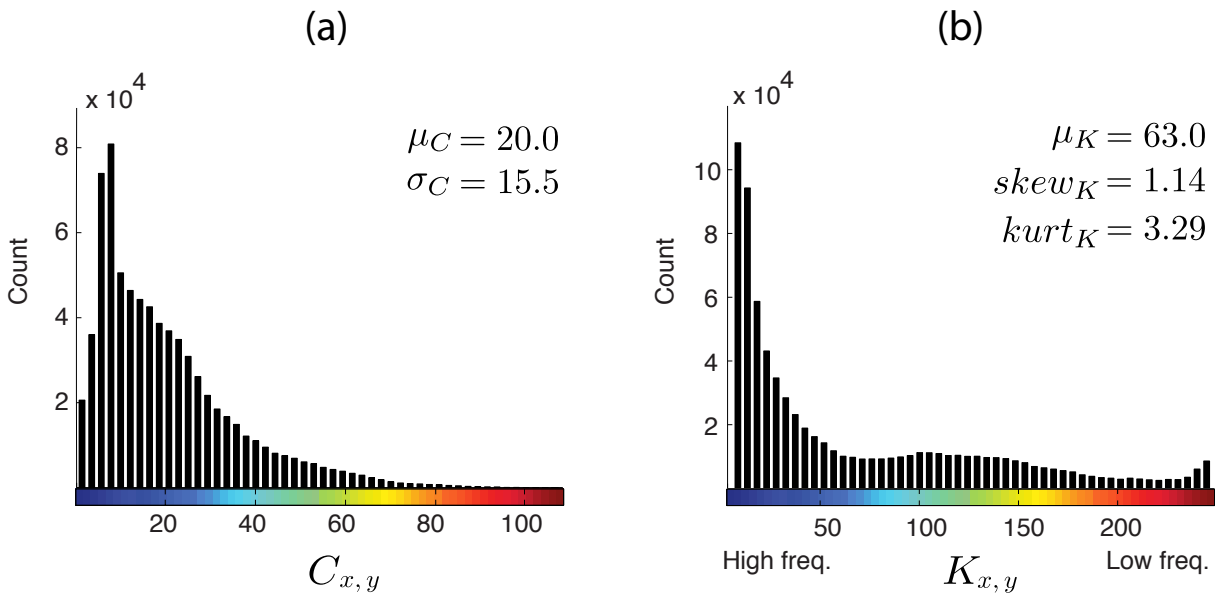

Figure 5.7: **Statistics of contrast and kurtosis maps**. (a) Histogram of the contrast map in Figure 5.6. (b) Histogram of the kurtosis map in Figure 5.6. The statistics of these p.d.f.s are presented at the top-right corner of the histograms. Colormaps for $C_{x,y}$ and $K_{x,y}$ values are preserved for easy understanding.

the maps and their histograms are useful for visual inspection and interpretation of the streetscape structure, they are not simple quantities. In other words, they can not be used directly as objective measures of the visual attributes of the streetscape.

The statistics of the maps on the other hand are quantities which describe very specific characteristics of the streetscape. Figure 5.7 shows the statistics of the contrast and kurtosis maps which are used in the proposed measure of complexity γ.

The first statistic is the mean μ_C of contrast values $C_{x,y}$, which is by definition a positive number. The mean value μ_C increases as the number of high-contrast regions increases. The second statistics is the standard deviation σ_C of contrast values $C_{x,y}$. σ_C can increase due to two factors. Firstly, it increases in the presence of image regions that generate contrast values $C_{x,y}$ higher than mean μ_C. On the other hand, it also increases with regions that yields contrast values lower than μ_C. In this way, σ_C represents the contrast "variety" in the streetscape image.

Regarding the kurtosis map, two statistics are used in the measure γ: the skewness $skew_K$ and the kurtosis $kurt_K$ of $K_{x,y}$ values (one must not confuse $K_{x,y}$ values with the kurtosis $kurt_K$ of their distribution).

Both skewness and kurtosis depend on the mean μ_K of the $K_{x,y}$ distribution. The skewness is generally regarded as a measure of asymmetry of a distribution in relation to its mean. For instance, if there is a tendency for $K_{x,y}$ values to be higher than the mean μ_K (i.e., the distribution is asymmetric towards its right-hand tail), then the skewness of the distribution is positive. On the other hand, in case distribution values tend to be lower than the mean, the skewness is negative. If the probability density distribution is symmetrical around its mean, the skewness is zero.

In Figure 5.7(b), the positive skewness, $skew_K = 1.14$, indicates asymmetry towards $K_{x,y}$ values higher than the mean $\mu_K = 63$. Notice that higher $K_{x,y}$ values represent lower

frequencies. Therefore, this positive $skew_K$ indicates asymmetry towards low-frequencies. In other words, there is a significant number of streetscape regions characterized by spatial frequencies lower than that represented by the mean μ_K.

For highly skewed distributions, however, it is important to investigate the presence of statistical outliers. These are generally defined as values *extremely* higher or lower than the mean of the distribution. For instance, in case of the histogram in Figure 5.7(b) with mean $\mu_K = 63$, outliers would be located at the extreme of the right-hand tail of the distribution.

Due to the properties of kurtosis, the magnitude of $kurt_K$ heavily reflects the presence of such values. Thus, $kurt_K$ is used in the denominator of measure γ to compensate $skew_K$ values which are high due to outliers in the $K_{x,y}$ distribution.

Figure 5.8 shows how the statistics of the contrast and kurtosis maps correlate with the subjective complexity rank r. In the scatter plot of Figure 5.8(a), the mean contrast μ_C is given in function of r-values. The correlation coefficient between μ_C and the subjective rank is R = 0.56.

The plot 5.8(b) shows the statistic σ_C. The correlation coefficient between σ_C and r is R = 0.57. Notice that the majority of nightscapes present lower μ_C and σ_C than dayscapes.

The positive correlations between μ_C, σ_C and the subjective rank indicate that complex streetscapes exhibit a higher number of objects or structures which elicit high changes of luminance and contrast in the scene (notice that RMS contrast is the standard deviation of luminance values).

The scatter plot 5.8(c) shows $skew_K$. This statistics has a correlation coefficient of R = 0.53 ($p < 10^{-5}$) with the subjective rank. This shows that the number of regions characterized by spatial frequencies lower than the mean in the streetscape tend to increase with complexity.

Figure 5.8(d) shows $kurt_K$. The correlation coefficient between $kurt_K$ and the subjective rank is R = 0.22, with a high p-value. This indicates that these variables are not significantly correlated. However, $kurt_K$ is an important statistic since it signalizes outliers in the $K_{x,y}$ distributions of the streetscapes.

The proposed measure γ is built as a direct combination of these observations on the characteristics of contrast and spatial frequency of streetscape scenes.

Objective rank analysis

This section shows how γ correlates with the subjective rank r given in Figure 5.4. Here, the following conventional measures are also analyzed: perimeter length, JPEG file size, subband entropy, feature congestion and Näsänen's measure.

Scatter plots in Figure 5.9 present the correlation behavior of the measures over the entire streetscape dataset. Figures 5.9(a) and 5.9(b) show the behavior of perimeter length and JPEG file size. (see Appendix E for parameter settings descriptions). These measures exhibit similar correlation coefficients with the subjective rank. In the scatter plots of both measures, nightscapes consistently receive lower values than dayscapes.

Figures 5.9(c) and 5.9(d) exhibit measures subband entropy and feature congestion. Fig. 5.9(e) and 5.9(f) shows the behavior of Näsänen's and the proposed γ. Notice that although these measures exhibit quite different correlation coefficients, they are also seem biased by nightscapes in same sense of the other shown measures. Still, the proposed γ exhibits the highest correlation when all streetscapes are considered ($R = 0.72$).

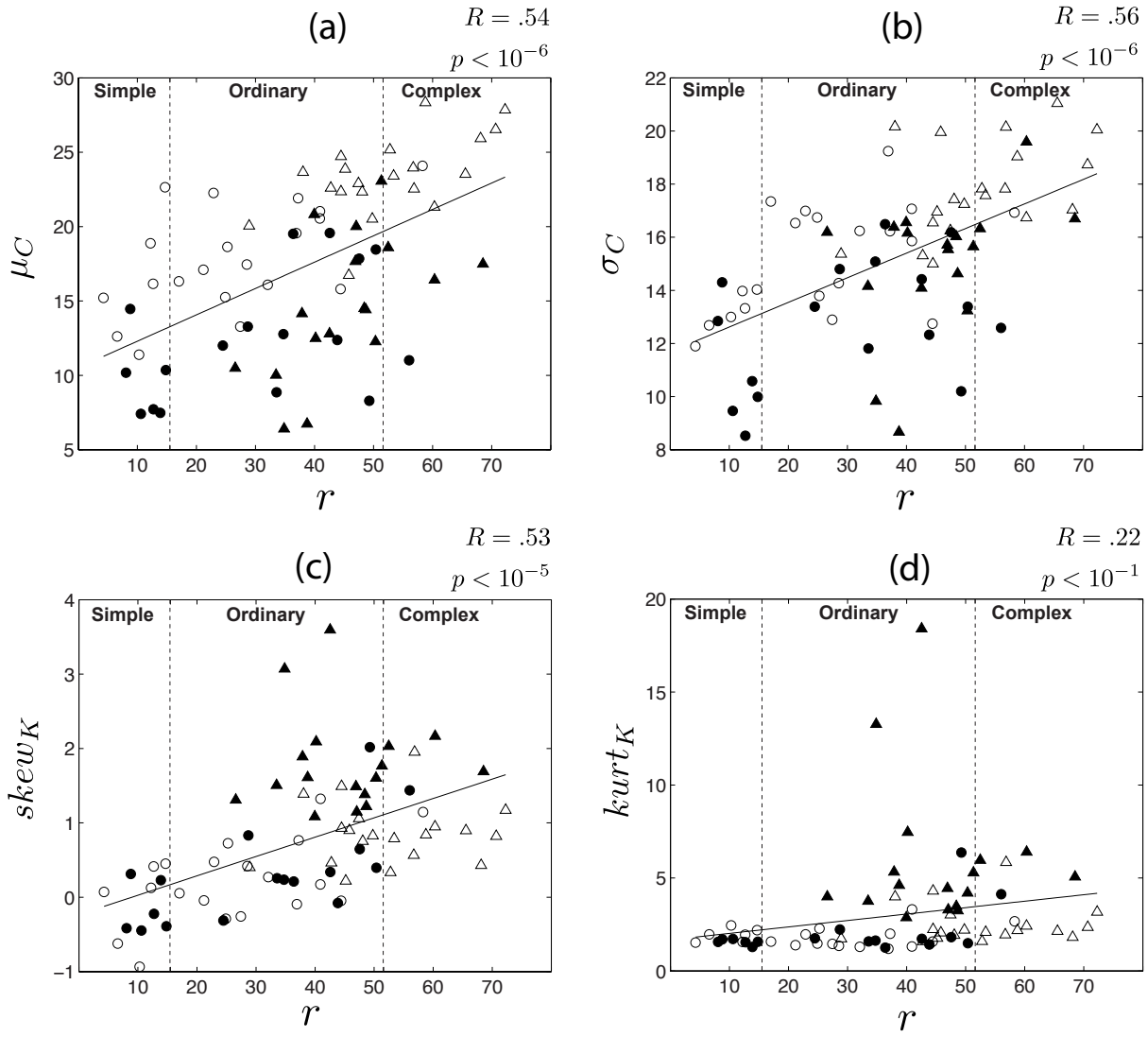

Figure 5.8: **Statistics of contrast and kurtosis maps (cont.)**. Statistics are given in function of the subjective rank r. (a) Mean contrast $\mu_{\mathbf{C}}$. (b) Standard deviation $\sigma_{\mathbf{C}}$ of contrast values. (c) Skewness of $K(x, y)$ values. (d) Kurtosis of $K(x, y)$ values. Vertical dotted lines represents the divisions between categories *simple*, *ordinary* and *complex*. In each scatter plot, the solid line represents the best least-squares-sense first-order polynomial fit. The numbers R and p at the top-right corner of each plot indicate the Pearson's correlation coefficient between objective and subjective ranks and its p-value, respectively.

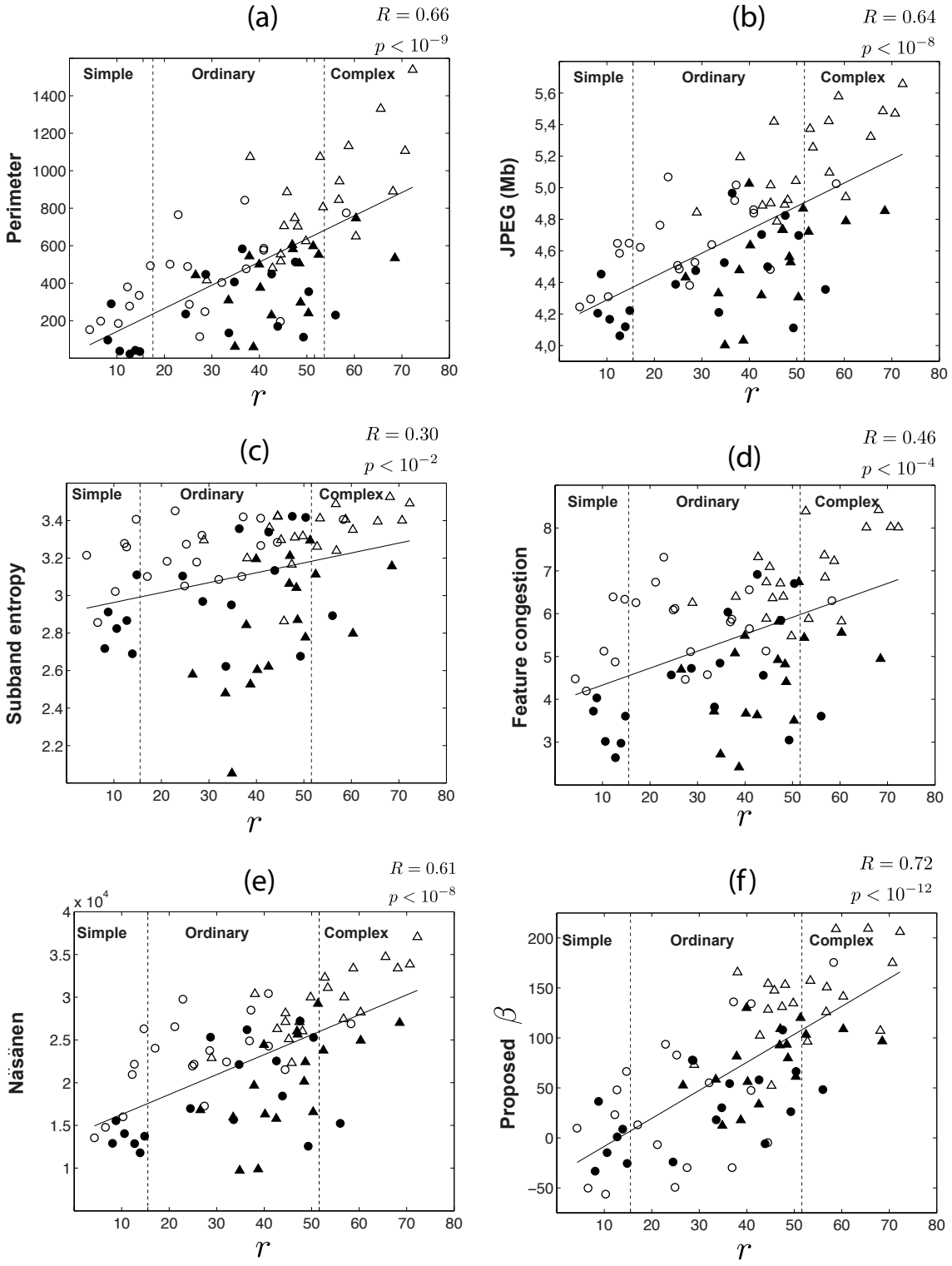

Figure 5.9: **Correlation between objective measures and subjective rank**. Objective measures are given in function of subjective rank r. (a) Perimeter length. (b) JPEG file size given in *megabytes*. (c) Subband entropy. (d) Feature Congestion. (e) Näsänen. (f) Measure γ. Correlation coefficient between objective and subjective ranks are given at the top right corner of each plot. Vertical dotted lines represents the divisions between categories *simple*, *ordinary* and *complex*. In each scatter plot, the solid line represents the best least-squares-sense first-order polynomial fit.

Table 5.1: **Correlation coefficients between objective and subjective ranks for individual streetscape types**. Correlation coefficients are calculated considering only the images in each type of streetscape. The types of streetscapes are shown in the most left column. Values in **bold** font represents significant correlation coefficients ($p < 0.001$).

	Perimeter	JPEG [97]	SB Entropy [39]	FC [39]	Näsänen [93]	Proposed γ [102]
Dayscapes	**0.79**	**0.83**	**0.53**	**0.65**	**0.80**	**0.77**
Nightscapes	0.59	**0.55**	0.27	0.46	**0.57**	**0.70**
Japan	**0.66**	**0.63**	**0.56**	**0.56**	**0.71**	**0.60**
Algeria	0.41	0.44	0.30	0.27	0.42	**0.51**
Simple	0.48	0.52	0.30	0.45	0.65	0.25
Ordinary	0.18	0.21	0.12	0.1	0.19	0.41
Complex	0.45	0.39	0.31	0.36	0.50	0.36

Table 5.1 exhibits the correlation coefficients when streetscape types are considered separately. From Table 5.1, it is clear that all objective measures have higher performance for daytime images. For instance, the correlation coefficient of JPEG file size is $R = 0.83$ for dayscapes and only $R = 0.55$ for nightscapes. On the other hand, the proposed γ exhibits the highest correlation for nightscapes, i.e., $R = 0.70$. Notice that γ also has the least variability between the correlation coefficients for dayscapes and nightscapes.

There are also variations in correlation for the other types of streetscapes. For instance, objective measures exhibit higher correlation coefficients for Japanese scenes than for those from Algeria. In this case, the proposed γ also exhibits less variation than other measures. For simple and complex scenes, Näsänen's measure is the most correlated with the subjective rank, i.e., $R = 0.65$ and $R = 0.50$, respectively. For ordinary category, γ has the highest correlation $R = 0.41$. Notice that for these three categories, p-values are higher than 0.001.

Discussion about contrast, spatial frequency and visual perception

Much has been understood about how the early visual system responds to contrast and spatial frequency. And while there is no established model of how these early responses influence the perception of complexity, it is interesting to consider physiological results that are related to γ (notice that for primitive shapes, response activity of visual cells increases with complexity [106]).

Local contrast can vary significantly within a visual scene [107]. A contrast map is a easy way to visualize this variation in terms of lower and higher contrast image areas. Now from a physiological point of view, it is important to understand how the response of early visual cells is influenced by these low and high contrast areas. Many studies have reported that in general firing rate of visual cells is not linearly related to the input contrast [108–110]. Specifically, firing rate increases linearly with contrast but reaches saturation at high contrast values. Furthermore, there are thresholds or contrast below which cells do not respond.

The measure γ is linearly related to the mean contrast of the streetscapes. Therefore, it increases with contrast but does not saturate as in the case of cell firing rate. Also, it does not account for any threshold effects. In order to mimic the physiological behavior, a proper non-linear transform would have to be applied on the contrast map in order to threshold and saturate contrast values.

The contrast sensitivity function (CSF) is another important physiological result that is related to the perception of spatial frequency [89]. The CSF defines how much contrast is necessary to perceive a spatial-frequency component. While the CSF can be different for each person, it generally shows that low-frequency components requires lower contrast to be perceived than high-frequency components. In other words, CSF shows that human subjects have higher sensitivity for low frequencies. Notice however that adaptation and masking effects during natural vision reduce this sensitivity after some period of exposure [111].

The contrast and kurtosis maps provide estimations of contrast and spatial frequency for each region within a scene. According to the CSF, spatial frequency components cannot be perceived in case image regions do not have the required minimum contrast. Since the current methodology does not account for the CSF, the frequency estimations for each image region may differ from what is actually perceived.

The discrepancy between what is measured and what is perceived could be significant specially for nighttime images due to lower luminance and contrast. In fact, it is known that visual acuity (i.e., the maximum perceived spatial frequency) is reduced in low luminance scenes [112]. Furthermore, changes in eye optics due to low luminance can introduce aberrations. These aberrations have the effect of decreasing the transmitted contrast for medium and high-spatial frequencies [113].

Changes in the distribution of light from daytime to nighttime also heavily influence the perception and interpretation of the "architectural" space [114]. Specifically, it is found dim light often results in shrinking the perceived size of objects, ornaments and the overall built environment. Unaccounted factors related to perception in low luminance and contrast might be the reason for the degraded performance of complexity measures in nightscapes.

The above are just few examples of issues related to the physiological processing and perception of contrast and spatial frequency. Notice that some of the complexity measures do not directly exploit these image properties. However, the characteristics of contrast and spatial frequency do influence the measurements in those methods. Furthermore, these methods are also strongly supported by knowledge about the early visual system.

The perimeter detection method, for example, is based on the number of edges detected in the scene. The process of edge detection is closely related to the filtering performed by the *simple* cells of the primary visual cortex (V1) [115, 116]. Specifically, these cells have very dedicated or specialized receptive fields. Due to this characteristic, simple cells have been primarily viewed as biological edge detectors [117, 118]. According to this, one could associate the perimeter detection measure to the activation of simple cells.

Further research, however, showed that the characteristics of V1 receptive fields could be artificially generated by *efficiently encoding* natural scenes [119]. In this coding process, filters are generated according to optimization functions whose goal is to maximize the amount of information extracted from the input signal. These results support a broader view of V1 cells where they are adapted to efficiently encode visual stimuli found in the environment [120].

Due to the nature of the signal filtering performed in the subband entropy method and in coding schemes such as JPEG, Rosenholtz suggests that these systems are likely to capture some of same information that is extracted by V1 cells [39]. Notice that the methodology used to compute our kurtosis map is also a V1-like filtering technique. However, in contrast to the subband entropy and JPEG filtering, the independent component

filters strongly focus on high-frequency bands.

In regard of JPEG filtering, it is also worthy noticing that there are additional constraints which are inspired by the human visual system. Specifically, the loss of information during coding is controlled so that low-frequency image components suffer less losses than high-frequency components. This rationale is derived from the human contrast sensitivity function.

After an image has been encoded by JPEG, the size of the digital file may be thought as the amount of information left in the image after losses. Similar thinking can be used to interpret the subband entropy measure. In this case, entropy represents the total amount of information in the frequency bands since there are no losses involved.

Interestingly, it has been shown that the size of the JPEG file is highly correlated with the number of edges in an image (i.e., the perimeter length measure) [98]. This result corroborates the connection between edge detection and coding of visual scenes. The correspondence between JPEG file size and the perimeter length measure can also be observed for streetscapes. As shown in Figure 5.9, these measures have quite close correlation coefficients with the subjective rank. Even analyzing at a streetscape type level (see Table 5.1), the maximum difference between their coefficients is not higher than 0.05. It is easy to see that this does not hold for any other pair of measures.

In this way, the measures of visual complexity analyzed here share similarities in terms of physiological foundations, image processing methodology, and correlation behavior with the subjective rank. In summary, these methods employ filtering techniques to extract low-level image characteristics which have well-understood influence in the human visual system. The objective measures are then derived in function of a single or a combination of these image characteristics.

Our results suggests that low-level image characteristics are indeed related to the complexity perceived in streetscapes. On daytime images for example, the use of these characteristics allow objective measures to be highly correlated with the opinion of the participants. Nonetheless, the effectiveness of these methods can considerably fluctuate across streetscape types. The same behavior is noticed for categories of images different from streetscapes [94, 98]. These studies suggest that different image characteristics may best suit different image categories.

In case of streetscapes, the statistics of local contrast and spatial frequency provide a competitive performance in comparison to the state-of-art methods. In fact, considering the entire dataset, the proposed measure γ exhibits the highest correlation with the subjective rank.

The measure γ has also less variability in correlation with subjective perception from daytime to nighttime images. For streetscapes, this is an important advantage. For instance, a more stable measure could be used to analyze visual interest and preference of pedestrians without requiring changes in the methodology. Furthermore, it could be used to analyze the human perception during nighttime driving, which has been pointed out as a difficult problem since the visual system behaves differently from daytime to nighttime [121].

5.4 Conclusion

This chapter has presented the method for measuring the perception of complexity in streetscapes. Our method is based on the use of the statistics of local contrast and spatial

frequency. The proposed method provides insight about the morphological features of the built space which are related to the perception of complexity. Specifically, in streetscapes high complexity is found correlated with the presence of high contrast structures and areas defined by spatial frequencies lower than the average in the scene. High contrast image features and the energy in low frequencies are in fact shown to drive human attention or emotional event processing [104, 105].

Now, the definition of streetscapes given in the introductory section clearly indicates that this category can hold very heterogeneous scenes. Diversity can come from many factors such as different types of architecture, geography, time scenario, and even season which directly influence the city vegetation.

Therefore, objective measures based on reduced sets of low-level image characteristics are unlikely to be satisfactory for all possible streetscapes. The statistical framework proposed in this work can be easily applied to identify new image characteristics related to the perception of complexity.

The diversity in this category also suggests that different perceptual mechanisms may engage during subjective evaluation of different streetscapes. As discussed before, the methods are still quite limited in accounting for such mechanisms. A proper implementation of perceptual related processes could improve objective measures with higher and more stable performance across different types of streetscapes.

The complexity perceived in streetscapes is known to influence important elements in urban life such as the visual interest of pedestrians and driving behavior. Here, one methodology is proposed for objectively measuring streetscape complexity based on the statistics of local contrast and spatial frequency. The proposed method exhibits higher correlations with subjective perception in comparison to conventional measures of complexity. Furthermore, it is found that this method is more effective and robust for nighttime scenes.

The proposed method also revealed structural features in streetscapes related to the perception of complexity. Specifically, it is found that higher complexity is associated with the presence of high contrast objects and image areas characterized by spatial frequencies lower that the average in the environment.

Chapter 6

Conclusion

6.1 Summary

Visual scene analysis is an old issue in engineering and information science. Generally, visual scene analysis is achieved by the implementation of two perceptual processes, i.e., segmentation and recognition. Segmentation and recognition are the base of many applications such as surveillance, medical image processing and assisted diagnostic, biometrics, object detection and tracking, visual inspection, text and document handling, etc. These applications are implemented in different platforms such as automotive, airborne and space, hospital and medical equipment, and general low-power consumer electronics such as smartphones and tablets. These platforms involve different trade-offs between energy consumption, time consumption, etc. In this regard, there is a need for methods which can fast perform segmentation and recognition with low computation.

This thesis has introduced a new methodology for visual scene analysis which can be used to implement both segmentation and recognition. Our methodology consists of analyzing the kurtosis of responses of independent component (IC) filters. In this way, the proposed methodology is mainly based on two concepts, which are independent component filters and kurtosis. Independent component filters are learned so as to have mutually statistically independent responses. Kurtosis is defined as the standardized fourth-order moment of a random variable.

In the proposed methodology, we calculate the responses of two different sets of IC filters to the same input image. Ideally, the first set of filters is designed to respond strongly to only one type or category of image defined a priori. Similarly, the second set of filters is designed to respond strongly to a type of image different than that of the first set of filters. By analyzing the kurtosis of the responses all filters, our methodology is able to segment or recognize different type of regions in the input image.

Our methodology is applied to three subjects or problems of visual scenes analysis. The first subject is *segmentation of depth-of-field image* (Chapter 3). The segmentation of depth-of-field is interesting for general amateur and professional image editing. Furthermore, the determination of depth-of-field is important for applications related to the enhancement of microscopy images. In this regard, then main challenges for methods of DOF segmentation is time consumption and computational complexity. The performance of the proposed method is analyzed for two different databases of depth-of-field images. And we use the classic F-measure as an objective criterion of performance. In this regard, our method exhibits the highest performance among the fast methods of segmentation of depth of field in terms of the F-measure. The performance of our method is only lower in comparison to that of time-consuming methods. Shortcomings of the proposed method

are the processing of object boundaries and in-focus low-frequency image areas.

The second subject of visual scene analysis is *segmentation of natural and man-made structures* (Chapter 4). The segmentation of natural and man-made structures is interesting for a number of applications including supporting autonomous driving and navigation, urban planning, target detection and tracking, etc. In this regard, examples of open problems for segmentation of natural and man-made structures are generalization and depth. We evaluate the performance of the proposed method for two different types of datasets. The first dataset consists of images acquired by still-devices. The second dataset consists of images acquired by moving devices. Each dataset consists of two different image databases. Here, we also use the F-measure as objective performance measure. Over the selected datasets, our proposed system exhibits the highest performance in terms of the F-measure in comparison to other methods.

The last subject is *measuring the perception of complexity in streetscapes* (Chapter 5). The perception of complexity in streetscapes influences the behavior of drivers and pedestrians. Therefore, measuring the perception of complexity has potential applications in the areas of assisted and autonomous driving, and urban planning. For evaluation of performance of our method, we have built a dataset of streetscape images. This dataset consists of images acquired in four cities. Two cities in Algeria and two cities in Japan. For evaluation of performance, we use the correlation coefficient between complexity ratings from human participants and the objective measures of complexity. In comparison to classic and new methods, the proposed measure of complexity exhibits higher correlation with the subjective rating of human participants. Also, the proposed method exhibits an important advantage. Specifically, our method can be used to measure the perception of complexity in nighttime images. In this way, our measure of complexity may became an interesting tool for studies of driving behavior during nighttime.

For all three subjects of visual scene analysis, it is important to highlight that the computational cost of our methodology is very low. Therefore, efficient implementations are very likely able to handle real-time applications. In this way, we conclude that this work is a relevant addition to science and engineering.

6.2 Applicability and limitations of the proposed methodology

Here, we discuss applicability or suitability, and the limitations of the proposed methodology. In our proposition, characteristics "A" and "B" are ideally mutually exclusive, i.e., an input image patch that exhibits characteristic "A" cannot exhibit "B". For real-world data, however, this ideal case may not occur often, i.e., both characteristics "A" and "B" appear in the input image at the same time. For example, let us assume that the goal of application is to determine if image patches contain natural or man-made structures. In this case, filters $\mathbf{a}_i$ are learned from natural scenes, filters $\mathbf{b}_i$ are learned from urban scenes. Consequently, low output kurtosis indicates that the input image patch contains a natural structure. High output kurtosis indicates that the input image patch contains a man-made structure. For a given scene, it is likely that many image patches contain both natural and man-made structure. When one of these image patches is the input of the proposed methodology, both sets of filters $\mathbf{a}_i$ and $\mathbf{b}_i$ may exhibit high-amplitude responses. In this case, the output kurtosis is low, because there is no response of amplitude *extremely* higher than the average amplitude value (i.e., all filter responses have

the similar amplitude). Therefore, since the output kurtosis is low, the methodology is going to (wrongly) report that the input image patch contains only an natural structure.

In our proposition, we also assume that the input image patch contains at least one of the characteristics "A" or "B". For real-world data, however, this ideal case may also not occur. If neither characteristic "A" nor vB" is present in the input image patch, or if the input contains a third, unexpected characteristic "C", filters $\mathbf{a}_i$ and $\mathbf{b}_i$ may not respond (i.e, respond with zero amplitude). In this case, the output kurtosis also low, because once again there are no responses of amplitude *extremely* higher than the average amplitude value. Therefore, the proposed methodology is going to fail.

Considering the above situations, the proposed methodology may exhibit poor performance or not be suitable at all. The applicability and limitations also depends on other properties of the methodology. For instance, the proposed methodology has only one layer of filters. This is an advantage in terms of processing speed and computational complexity. However, this can also be a disadvantage in case of complex or higher-order recognition problems. For example, if characteristic "A" or "B" require many layers of filtering to be recognized, the proposed methodology is likely to fail (because it is based on only one layer of filters). One example of complex recognition problem is rotational invariant face recognition. For this application, a method based on deep neural networks has been recently proposed [11]. That method exhibits a high performance, but it requires a high number of filtering layers.

Another issue of the proposed methodology is the relation between the number of filters $\mathbf{a}_i$ and that of $\mathbf{b}_i$. As described in section 2.2, the number of filters $\mathbf{a}_i$ is p, the number of filters $\mathbf{b}_i$ is q, and $p \gg q$. The condition $p \gg q$ imposes a constraint of *dimensionality* on characteristics "A" and "B". Specifically, if $p \gg q$, then characteristic "B" should require far less filters to be detected in the input image patch than characteristic "A".

For the case of segmentation of natural and man-made structures, we have used 255 filters $\mathbf{a}_i$ learned from nature scenes, and 4 filters $\mathbf{b}_i$ learned from urban scenes. Thus, we have assumed that man-made structures require less filters to be detected than natural structures. Our assumption is based on studies of dimensionality of the space of natural scenes. Specifically, it has been shown that the average logarithmic euclidean distance between different image regions in nature scenes is far smaller than that in Gaussian images [122]. This suggests that nature scenes have lower dimensionality than that of Gaussian images. In this regard, it can be shown that the average logarithmic euclidean distance between image regions in urban scenes is far lower than that in nature scenes. This suggests that urban scenes should have far lower dimensionality than that of nature scenes. Therefore, man-made structures should require far less filters to be detected than natural structures.

In fact, for several subjects in the field of pattern recognition, research suggests that different categories of signals indeed have different dimensionality [123]. For instance, it is known that fractal dimension and entropy of heart rate time series is consistently different among humans with different cardiac conditions. Furthermore, many works in pattern recognition perform a pre-processing step generally called as *dimensionality reduction*. In dimensionality reduction, data components or dimensions are permanently excluded from the bulk of data according to some criterion. In this thesis, dimensionality reduction is also performed in order to make the number of filters $\mathbf{a}_i$ far higher than the number of filters $\mathbf{b}_i$, i.e., $p \gg q$. However, the downside of dimensionality reduction is that it can exclude information which may prove to be important at some point. For

instance, in the proposed method for segmentation of natural and man-made structures, 252 filters learned from urban scenes are excluded. These excluded filters may contain information necessary to detect specific man-made structures. Therefore, dimensionality reduction can limit the performance of the proposed methodology.

Another issue that must be addressed in this methodology is the choice of methods used for learning filters $\mathbf{a}_i$ and $\mathbf{b}_i$, i.e., independent component analysis. We have chosen independent component analysis because it is a powerful technique for (blind) source separation. Our rationale is that the problem of image segmentation may be understood from a source-separation point of view. For example, different image partitions may be thought as different sources. Thus, different subsets of IC filters may help "separate" or segment different image partitions. However, it is important to notice that other techniques may also be used. For instance, one may choose to use principal component filters, or even image templates similar to template matching systems. The proposed methodology is general in this regard. Furthermore, it is likely that the best choice of filters depends on the type of application.

The last issue is the choice of kurtosis. As we have discussed before, kurtosis is used to discriminate or distinguish different responses patterns. We have chosen to use kurtosis because kurtosis is generally used to characterize the response activity of cortical cells. Specifically, the responses patterns of cortical cells can exhibit a property know as *sparseness* [124, 125]. Under a sparse response regime, "a small subset within a neural population will respond strongly to a stimulus, while most will respond poorly" [124]. The methodology for visual scene analysis proposed in this thesis is based on such knowledge from biological systems.

Depending on the application, other statistics such as variance and skewness may also be suitable. By using the variance of response patterns instead of the kurtosis, the methodology would indicate the presence of responses either lower or higher than the average response value. This does not seem useful for discrimination or recognition applications. By using the skewness, however, the methodology would indicate that the probability distribution of the response pattern is asymmetrical. In this case, negative and positive skewed response patterns could be discriminated. This seems useful for recognition problems.

In this regard, there are actually other measures which are similar to kurtosis. For instance, we can cite *negentropy*. Both kurtosis and negentropy are associated with the idea of measuring sparseness and non-Gaussianity. However, we have chosen to use kurtosis because the computational cost of calculating negentropy seems higher than that of calculating kurtosis.

6.3 Future work

Future work can focus on analyzing new techniques for creating or learning filters. In fact, there are techniques closely related to independent component analysis such as *sparse coding* [119] and *topographic independent component analysis* [51]. One may even analyze the use *eigen*-images or image templates as filters. Also, it is interesting to explore the use of statistical measures different from kurtosis for discriminating response patterns, given that the tradeoff between performance and computational cost is in view.

In a related note, the current architecture of the methodology which consists of a single layer of filters could be modified into a hierarchical architecture. In this new

architecture, the output kurtosis value from a previous layer would activate the next layer in the hierarchy, allowing further filtering and recognition of more complex visual characteristics.

In another line of research, future work can focus on developing efficient implementations of the methodology for real-time applications. For instance, a *hardware* implementation for identification of man-made structures in aerial images could be easily developed if based on our methodology. Also, one can implement and apply our measure of visual complexity on studies of walking and driving behavior. Finally, one may apply the proposed methodology for other subjects or problems of visual scene analysis.

Acknowledgments

I would like to thank the Ministry of Education, Culture, Sports, Science and Technology of Japan for giving me the opportunity to study in Japan. Furthermore, I would like to thank Nagoya University for accepting me as its student. I am always going to behold Nagoya University as a place of excellency, truth and hard work.

I would like to thank Professor Noboru Ohnishi from the Graduate School of Information Science, Nagoya University. I have used the teachings of Professor Ohnishi not only in academy, but in many aspects of my life. I would like to thank him for the patience and tolerance towards my mistakes. For him, my eternal gratitude for giving me guidance in such critical time of my life.

I would like to thank Professor Yoshinori Takeuchi from the Department of Information Systems, Daido University. Professor Takeuchi has also intensively guided me in my professional and daily life in Japan. I would like to thank him for being present from the moment I arrived in the Nagoya airport.

I would also like to thank Professor Hiroaki Kudo and Professor Tetsuya Matsumoto from the Graduate School of Information Science, Nagoya University. Professors Kudo and Matsumoto have always shown kindness and support.

I would like to thank Professor Naoji Matsumoto and Researcher Ahmed Mansouri from the Department of Architecture, Nagoya Institute of Technology. Their knowledge and support was extremely important to this research.

I would like to thank Professor Allan Kardec Barros from the Laboratory of Biological Information Processing, Universidade Federal do Maranhão, Brazil. Professor Allan has continuously guided and supported me since my undergraduate student days.

I would like to thank Professor Hiroshi Murase from the Graduate School of Information Science, Nagoya University, for reviewing my thesis and work.

I would like to thank and acknowledge that this research was partially funded by The Hori Sciences and Arts Foundation.

I would like to thank Katiuscia Horita for continuously giving me great kindness and support to this date.

A special thanks to my important friends Fausto Oliveira, Lemya Kacha Epe Mansouri, Saulo de Oliveira, Diogo Santos, Itzel Santos, Mauricio Kugler, Celso Sakuraba, Victor Benso, Lucas Malta, Watchareeruetai Ukrit.

Appendix A

Derivation of FastICA algorithm

The derivation of FastICA algorithm is fully described at [51,69]. This appendix includes the derivation of the main adaptation rule of FastICA. In Chapter 2, we have defined equation

$$y_i = \mathbf{h}_i^{\mathrm{T}} \mathbf{z}, \tag{A.1}$$

where elements z_i of vector $\mathbf{z}$ are mutually uncorrelated. In the FastICA algorithm, vector $\mathbf{h}_i$ is adapted so as to maximize the negentropy of y_i. The negentropy of y_i is represented here as $J(y_i)$, and it is approximated by

$$J(y_i) \approx [E\{G(y_i)\} - E\{G(v)\}]^2, \tag{A.2}$$

where v represents a Gaussian random variable of zero mean and unit variance, $G(\cdot) = \log\cosh(\cdot)$, and $E\{\cdot\}$ is the expectation operator.

For the FastICA algorithm, the maxima of negentropy is found at a certain optima of $E\{G(\mathbf{h}_i^{\mathrm{T}}\mathbf{z})\}$ under the constraint that $\|\mathbf{h}_i\|^2 = 1$. According to Kuhn-Tucker conditions [126], such optima is obtained at points where

$$E\{\mathbf{z}g(\mathbf{h}_i^{\mathrm{T}}\mathbf{z})\} - \beta\mathbf{h}_i = 0, \tag{A.3}$$

where $g(\cdot) = \tanh(\cdot)$ is the derivative of $G(\cdot) = \log\cosh(\cdot)$ and β is a constant. Eq. (A.3) can be solved by using the Newton method. Let us denote the left-hand side of Eq. (A.3) as F and take its Jacobian matrix as

$$\frac{\partial F}{\partial \mathbf{h}_i} = E\{\mathbf{z}\mathbf{z}^{\mathrm{T}} g'(\mathbf{h}_i\mathbf{z})\} + \beta\mathbf{I}. \tag{A.4}$$

Notice that $\mathbf{z}$ consists of uncorrelated elements. Thus, $E\{\mathbf{z}\mathbf{z}^{\mathrm{T}} g'(\mathbf{h}_i\mathbf{z})\}$ is approximated as $E\{\mathbf{z}\mathbf{z}^{\mathrm{T}} g'(\mathbf{h}_i\mathbf{z})\} \approx E\{\mathbf{z}\mathbf{z}^{\mathrm{T}}\}E\{g'(\mathbf{h}_i\mathbf{z})\} = E\{g'(\mathbf{h}_i\mathbf{z})\}\mathbf{I}$. Based on this new Jacobian matrix, the following approximative Newton iteration is considered

$$\mathbf{h}_i \leftarrow \mathbf{h}_i - [E\{\mathbf{z}g(\mathbf{h}_i^{\mathrm{T}}\mathbf{z})\} + \beta\mathbf{h}_i]/[E\{g'(\mathbf{h}_i^{\mathrm{T}}\mathbf{z})\} + \beta]. \tag{A.5}$$

By multiplying the right side of (A.5) by $\beta + E\{g'(\mathbf{h}_i^{\mathrm{T}}\mathbf{z})\}$, we have that

$$\mathbf{h}_i \leftarrow E\{\mathbf{z}g(\mathbf{h}_i^{\mathrm{T}}\mathbf{z})\} - E\{g'(\mathbf{h}_i^{\mathrm{T}}\mathbf{z})\}\mathbf{h}_i. \tag{A.6}$$

This is the main adaptation rule of FastICA algorithm.

Appendix B

Kurtosis for known distributions

B.1 Gaussian distribution

The Gaussian probability distribution can be defined as

$$p(u) = \frac{1}{\sqrt{2\pi}} e^{-\frac{u^2}{2}}, \tag{B.1}$$

where $u \in \mathbb{R}$ represents a realization of a random variable U. Here, assume that U has zero mean and unit standard-deviation.

The fourth central moment of U is represented here as μ_4, and it is calculated as

$$\begin{aligned} \mu_4 &= \tfrac{1}{\sqrt{2\pi}} \textstyle\int_{-\infty}^{\infty} u^4 e^{-\frac{u^2}{2}} dx \\ &= \tfrac{2}{\sqrt{2\pi}} \textstyle\int_{0}^{\infty} u^4 e^{-\frac{u^2}{2}} dx \end{aligned} \tag{B.2}$$

Making $y = u^2$, (B.2) becomes

$$\begin{aligned} \mu_4 &= \tfrac{1}{\sqrt{2\pi}} \textstyle\int_0^{\infty} y^{\frac{3}{2}} e^{-\frac{y}{2}} dx \\ &= \tfrac{1}{\sqrt{2\pi}} \{-2y^{\frac{3}{2}} e^{-\frac{y}{2}}|_0^{\infty} + 3 \textstyle\int_0^{\infty} y^{\frac{1}{2}} e^{-\frac{y}{2}} dy\} \\ &= \tfrac{1}{\sqrt{2\pi}} \{-2y^{\frac{3}{2}} e^{-\frac{y}{2}}|_0^{\infty} + 3[-2y^{\frac{1}{2}} e^{-\frac{y}{2}}|_0^{\infty} + 2 \textstyle\int_0^{\infty} e^{-\frac{u^2}{2}} du]\} \\ &= \tfrac{1}{\sqrt{2\pi}} (-2y^{\frac{3}{2}} e^{-\frac{y}{2}} - 6y^{\frac{1}{2}} e^{-\frac{y}{2}})|_0^{\infty} + 3 \\ &= 3. \end{aligned} \tag{B.3}$$

B.2 t-distribution

t-distribution can be defined as

$$p(u) = c\left(1 + \frac{u^2}{n}\right)^{-\frac{n+1}{2}}. \tag{B.4}$$

The constant c is defined as

$$c = \frac{1}{\sqrt{n}} \frac{1}{B(\frac{n}{2}, \frac{1}{2})}, \tag{B.5}$$

where B is the *Beta* function, and $n \in \mathbb{N}$ is generally named degrees of freedom.

The k-th moment of t-random variable U is defined as

$$\begin{aligned} \mu_k &= \textstyle\int_{-\infty}^{\infty} u^k p(u) du \\ &= \textstyle\int_{-\infty}^{0} u^k p(u) du + \int_{0}^{\infty} u^k p(u) du. \end{aligned} \tag{B.6}$$

Making $y = -u$ in the first integral of (B.6),

$$\begin{aligned}\mu_k &= -\int_{-\infty}^{0}(-y)^k p(-y)dy + \int_0^{\infty} u^k p(u)du \\ &= (-1)^k \int_0^{\infty} y^k p(-y)dy + \int_0^{\infty} u^k p(u)du.\end{aligned} \tag{B.7}$$

Since $p(-u) = p(u)$, (B.7) can be written as

$$\mu_k = (1 + (-1)^k) \int_0^{\infty} u^k p(u)du. \tag{B.8}$$

Given (B.4), the integral $\int_0^{\infty} u^k p(u)du$ in (B.8) can be written as

$$\begin{aligned}&\int_0^{\infty} u^k p(u)du \\ &= c\int_0^{\infty} u^k \left(1 + \frac{u^2}{n}\right)^{-\frac{1}{2}(n+1)} du\end{aligned} \tag{B.9}$$

Making $t = \frac{u^2}{n}$, (B.9) becomes

$$\begin{aligned}&= c\int_0^{\infty}(nt)^{\frac{k}{2}}(1+t)^{\left(-\frac{n}{2}-\frac{1}{2}\right)}\frac{\sqrt{n}}{2}\frac{1}{\sqrt{t}}dt \\ &= c\tfrac{1}{2}n^{\frac{k+1}{2}}\int_0^{\infty} t^{\left(\frac{k}{2}-\frac{1}{2}\right)}(1+t)^{\left(-\frac{n}{2}-\frac{1}{2}\right)}dt \\ &= c\tfrac{1}{2}n^{\frac{k+1}{2}}B\left(\tfrac{k+1}{2}, \tfrac{n-k}{2}\right)\end{aligned} \tag{B.10}$$

Given the definition of c in (B.5), (B.10) can be written as

$$\begin{aligned}&= \tfrac{1}{2}\frac{1}{\sqrt{n}}\frac{1}{B\left(\frac{n}{2},\frac{1}{2}\right)}n^{\frac{k+1}{2}}B\left(\tfrac{k+1}{2}, \tfrac{n-k}{2}\right) \\ &= \tfrac{1}{2}n^{\frac{k}{2}}\left[\frac{\Gamma\left(\frac{n}{2}\right)\Gamma\left(\frac{1}{2}\right)}{\Gamma\left(\frac{n+1}{2}\right)}\right]^{-1} \\ &\quad \cdot\frac{\Gamma\left(\frac{k+1}{2}\right)\Gamma\left(\frac{n-k}{2}\right)}{\Gamma\left(\frac{n+1}{2}\right)} \\ &= \tfrac{1}{2}n^{\frac{k}{2}}\frac{\Gamma\left(\frac{k+1}{2}\right)\Gamma\left(\frac{n-k}{2}\right)}{\Gamma\left(\frac{n}{2}\right)\Gamma\left(\frac{1}{2}\right)}.\end{aligned} \tag{B.11}$$

Finally, given (B.8) and (B.11), the k-th moment of U is

$$\mu_k = \begin{cases} n^{\frac{k}{2}}\dfrac{\Gamma\left(\frac{k+1}{2}\right)\Gamma\left(\frac{n-k}{2}\right)}{\Gamma\left(\frac{n}{2}\right)\Gamma\left(\frac{1}{2}\right)}, \text{if } k \text{ is even} \\ 0, \text{if } k \text{ is odd.}\end{cases} \tag{B.12}$$

Based on the above equation, kurtosis is $\frac{\mu_4}{(\mu_2)^2} = 9$.

Appendix C

Analyzing differences between IC filters and Gabor functions

In section 3.3, we have shown that there are differences between IC filters and Gabor functions. These differences can be observed by comparing the average 2D Fourier amplitude spectrum of IC filters to that of Gabor functions. In this regard, we have shown that for low spatial frequencies, isolines of the amplitude spectrum of IC filters have a "diamond" shape. The isolines of the amplitude spectrum of Gabor functions have a more circular shape. This difference has important implications to neuroscience, specifically to the area of modeling the receptive fields of primary visual cortex (V1)cells. Furthermore, there are implications to the field of image coding. Here, we discuss these implications.

C.1 Biological background

The receptive fields of V1 cells are filters used by the brain for processing visual stimuli [115]. It is known that V1 receptive fields can be modeled by IC filters or Gabor functions [51, 119, 127]. Furthermore, the shape of V1 receptive fields adaptively change in function of the input visual stimuli [128]. The goal of this adaptation is to enhance transmission of information in the cortex.

The adaptation of receptive fields occur by many mechanisms. One example of adaptation mechanism is *amplitude compensation* [128]. In this mechanism, the average Fourier amplitude spectrum of receptive fields change and become similar to the "inverse" of the amplitude spectrum of the visual stimulus. For example, let's assume that the Fourier amplitude spectrum of the visual stimulus has shape in the form $\frac{1}{f}$, where f is frequency. In this case, amplitude spectrum of the receptive field should become similar to f (the inverse of $\frac{1}{f}$). This is mechanism is similar to the whitening process.

C.2 Experiments and results

Amplitude compensation when visual stimuli are natural images

In this section, let us analyze the case of amplitude compensation when the visual stimuli are natural images. For this analysis, the first step is to compute the Fourier amplitude spectrum of natural images. Thus, we have selected 100.000 image patches of 16×16 pixels from natural scenes of the McGill Calibrated Color Image Database [68].

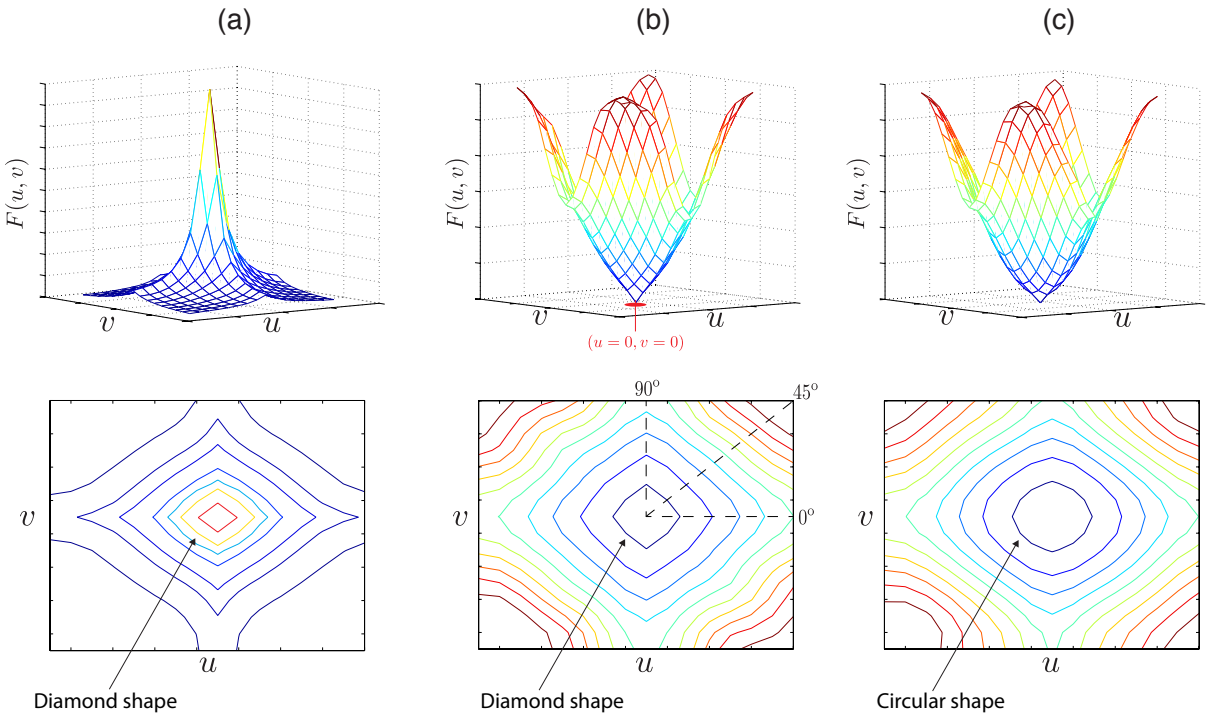

Figure C.1: **Average 2D Fourier amplitude spectra**.(a) Average amplitude spectrum of natural images. (b) Average amplitude spectrum of IC filters. (c) Average amplitude spectrum of Gabor functions. Plots (b) and (c) are originally from Figure 3.5. The plots at the bottom show the profile curves or "isolines" for different levels of amplitude. Notice that towards low frequencies, the amplitude spectrum of natural images and that of IC filters have isolines with "diamond" shape.

Then, we have computed the 2D Fourier amplitude spectrum of each of the 100.000 image patches. Over all 100.000 2D amplitude spectra, we then compute the average. This average Fourier amplitude spectrum is shown in Figure C.1(a). Notice that the amplitude of natural images decreases with frequency, i.e., their amplitude spectrum has form of $\frac{1}{f}$. Furthermore, notice that for a fixed spatial frequency, amplitude is higher at normal orientation (0 degrees and 90 degrees) than that at oblique orientations (45 degrees and 135 degrees). This difference in amplitude for normal and oblique orientations is the reason why isolines of the amplitude spectrum have "diamond" shape. The "diamond" shape of isolines is a long known property of natural images (See reference [79] for a review).

According to the amplitude compensation mechanism [128], after visual stimulation with natural images, the average amplitude spectra of V1 receptive fields should become similar to the inverse of the amplitude spectrum of natural images. In other words, the amplitude in V1 receptive fields should increase with frequency, i.e., amplitude spectrum of form f (the inverse of $\frac{1}{f}$).

Furthermore, amplitude compensation introduces another effect. Specifically, for a fixed spatial frequency, the amplitude of V1 receptive fields should become lower at normal orientations than that at oblique orientations (this is the inverse of amplitude pattern of the input). Therefore, the average amplitude spectrum of V1 receptive fields must also exhibit isolines with "diamond" shape.

Now, since IC filters and Gabor functions are models of V1 receptive fields, they should also execute *amplitude compensation*. As shown in Figures C.1(a) and (b), the average amplitude spectra of IC filters and Gabor functions have the form of f, i.e, amplitude increases with frequency. However, at low spatial frequencies, only the amplitude spectrum of IC filters have isolines with "diamond" shape. Therefore, at low spatial frequencies, IC filters offer better amplitude compensation than that of Gabor functions.

Image coding using IC filters and Gabor functions

Let us define the following experiment of image coding. Assume that IC filters calculated by the FastICA algorithm are represented by vectors $\mathbf{w}_i \in \mathbb{R}^k$. Assume that each $\mathbf{w}_i$ is a row of matrix $\mathbf{W} \in \mathbb{R}^k \times \mathbb{R}^k$. Here, the inverse of $\mathbf{W}$ is represented by matrix $\mathbf{Q} \in \mathbb{R}^k \times \mathbb{R}^k$. The columns of $\mathbf{Q}$ are represented by vectors $\mathbf{q}_i \in \mathbb{R}^k$. Vectors $\mathbf{q}_i$ are generally called independent component basis functions.

Furthermore, let us assume that vector $\mathbf{n} \in \mathbb{R}^k$ represents an image read in raster-scan fashion. Now, let's define vector $\mathbf{n}_q \in \mathbb{R}^k$ as the sum of IC basis functions, i.e.,

$$\mathbf{n}_q = c'_1\mathbf{q}_1 + c'_2\mathbf{q}_2 + \cdots + c'_k\mathbf{q}_k. \tag{C.1}$$

where coefficients $c'_i \in \mathbb{R}$ are calculated as follows. Firstly, a vector $\mathbf{c} = [c_1, c_2, \ldots, c_k]^{\mathrm{T}}$ is computed as

$$\mathbf{c} = [\mathbf{Q}\mathbf{Q}^{\mathrm{T}}]^{-1}[\mathbf{Q}\mathbf{n}]. \tag{C.2}$$

The coefficients c'_i in Eq. C.1 are a quantized version of c_i. Let's define $n_i \in \mathbb{R}$ and $n_{iq} \in \mathbb{R}$ as the i-th elements of vectors $\mathbf{n}$ and $\mathbf{n}_q$, respectively. Finally, let's define the error

$$e_q = \frac{1}{k}\sum_{i=1}^{k}(n_i - n_{iq})^2. \tag{C.3}$$

Similarly, assume that Gabor functions estimated by fitting of IC filters are represented by vectors $\mathbf{g}_i \in \mathbb{R}^k$. Assume that each $\mathbf{g}_i$ is a row of matrix $\mathbf{G} \in \mathbb{R}^k \times \mathbb{R}^k$. Here, the inverse of $\mathbf{G}$ is represented by matrix $\mathbf{P} \in \mathbb{R}^k \times \mathbb{R}^k$. The columns of $\mathbf{P}$ are represented by vectors $\mathbf{p}_i \in \mathbb{R}^k$. Here, let us call vectors $\mathbf{p}_i$ as Gabor basis functions.

Now, let's define vector $\mathbf{n}_p \in \mathbb{R}^k$ as the sum of Gabor basis functions, i.e.,

$$\mathbf{n}_p = v'_1\mathbf{p}_1 + v'_2\mathbf{p}_2 + \cdots + v'_k\mathbf{p}_k. \tag{C.4}$$

where coefficients $v'_i \in \mathbb{R}$ are calculated as follows. Firstly, a vector $\mathbf{v} = [v_1, v_2, \ldots, v_k]^{\mathrm{T}}$ is computed as

$$\mathbf{v} = [\mathbf{P}\mathbf{P}^{\mathrm{T}}]^{-1}[\mathbf{P}\mathbf{n}]. \tag{C.5}$$

The coefficients v'_i in Eq. C.4 are a quantized version of v_i. Let's define $n_{ip} \in \mathbb{R}$ as the i-th element of vector $\mathbf{n}_p$, respectively. Finally, let's define the error

$$e_p = \frac{1}{k}\sum_{i=1}^{k}(n_i - n_{ip})^2. \tag{C.6}$$

In this way, e_q and e_p represent the reconstruction error of image $\mathbf{n}$ when using IC and Gabor basis functions, respectively. Notice that over many images $\mathbf{n}$, one can compute the average of e_q and e_p. Here, we have calculated these averages over 20.000 images patches. The average of e_q and e_p are shown in Figure C.2 in terms of signal-to-distortion (SDR) ratio. The coefficients c_i and v_i were quantized using 7 bits. Figure C.2 clearly shows that IC basis functions generates higher SDR than that of Gabor basis functions.

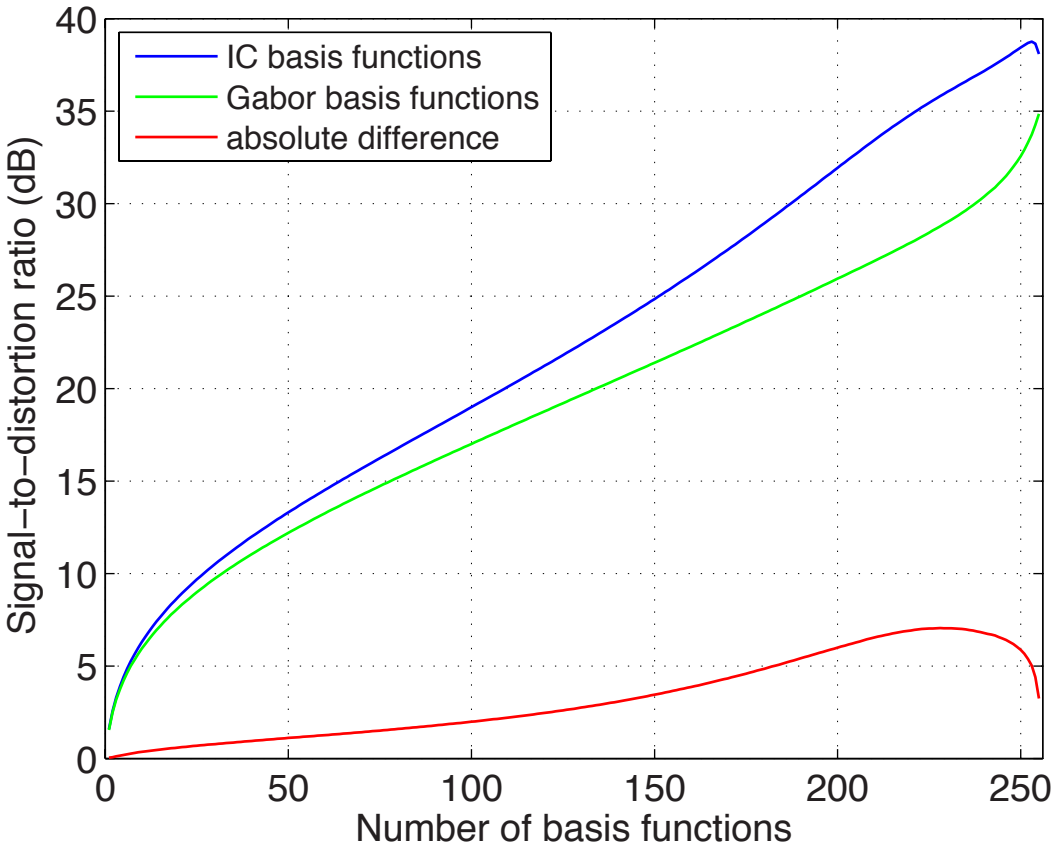

Figure C.2: **Average signal-to-distortion ratio when coding natural images.** (Blue) IC basis functions. (Green) Gabor basis functions. (Red) Absolute difference between (Red) and (Blue).

C.3 Conclusion

Here, we have presented an analysis of the differences between IC filters and Gabor functions. Firstly, it is shown that IC filters offer better *amplitude compensation* for low spatial frequencies than that of Gabor functions. Secondly, it is shown that IC basis functions generates higher signal-to-distortion ratio for image coding than that of Gabor basis functions. Therefore, these differences are relevant to the fields of neuroscience and image coding.

Future work should focus on modeling the difference between IC filters and Gabor functions. For instance, one can assume that an IC filter $\mathbf{w}_i$ can be described as

$$\mathbf{w}_i = \mathbf{g}_i * \mathbf{h}_i + \mathbf{s}_i, \tag{C.7}$$

where $\mathbf{g}_i$ is a Gabor function, $\mathbf{h}_i$ is an unknown function that deforms the Gabor component $\mathbf{g}_i$, and $\mathbf{s}_i$ is an additive noise signal of unknown probability distribution. Therefore, $\mathbf{h}_i$ and $\mathbf{s}_i$ in Eq. C.7 describe the difference between IC filters and Gabor functions.

By modeling the differences between IC filters and Gabor functions and extending our research, one might be able to provide better models of V1 receptive fields, and also more efficiently computer methods for encoding natural images.

Appendix D

Segmentation of natural and man-made structures based on mean-squared error

This appendix describes our methodology for segmentation of natural and man-made structures using mean square error [84].

Assume that IC filters calculated by the FastICA algorithm are represented by vectors $\mathbf{w}_i \in \mathbb{R}^k$. Assume that each $\mathbf{w}_i$ is a row of matrix $\mathbf{W} \in \mathbb{R}^k \times \mathbb{R}^k$. Here, the inverse of $\mathbf{W}$ is represented by matrix $\mathbf{Q} \in \mathbb{R}^k \times \mathbb{R}^k$. The columns of $\mathbf{Q}$ are represented by vectors $\mathbf{q}_i \in \mathbb{R}^k$. These vectors are generally called independent component basis functions.

Let us define vector $\mathbf{n}' \in \mathbb{R}^k$ as the sum of IC basis functions, i.e.,

$$\mathbf{n}' = c_1\mathbf{q}_1 + c_2\mathbf{q}_2 + \cdots + c_k\mathbf{q}_k. \tag{D.1}$$

For $c_i \in \mathbb{R}$, vector $\mathbf{c}_i = [c_1, c_2, \ldots, c_k]^{\mathrm{T}}$ is computed as

$$\mathbf{c}_i = [\mathbf{Q}\mathbf{Q}^{\mathrm{T}}]^{-1}[\mathbf{Q}\mathbf{n}], \tag{D.2}$$

where $\mathbf{n} \in \mathbb{R}^k$. Based on $\mathbf{n}$ and $\mathbf{n}'$ let us define the error

$$e = \frac{1}{k}\sum_{i=1}^{k}(n_i - n_i')^2, \tag{D.3}$$

where n_i and n_i' are the i-th element of vectors $\mathbf{n}$ and $\mathbf{n}'$, respectively.

Now, let us assume that vector $\mathbf{n}$ represents an image patch read in raster scan fashion. Also, $\mathbf{Q}_{nat}$ and $\mathbf{Q}_{man}$ represent basis functions learned from natural and man-made scenes, respectively. Finally, assume that e_{nat} and e_{man} represent the error in Eq. (D.3), when basis functions $\mathbf{Q}_{nat}$ and $\mathbf{Q}_{man}$ are used respectively.

In this way, our hypothesis is as follows. If $e_{nat} < e_{man}$, the image patch $\mathbf{n}$ contains a natural structure. Else, i.e., $e_{nat} > e_{man}$, the image patch contains a man-made structure.

Appendix E

Setting algorithm parameters for measuring the perception of complexity in streetscapes

Parameter settings

The methods used for comparison (i.e., perimeter length, JPEG file size, subband entropy and feature congestion) are based on image processing flows which require input settings. In order to determine these settings, an analysis is carried out for each method. Specifically, input settings are varied so as to maximize the correlation of objective measures with the subjective complexity rank. The methods are only considered for comparison when using settings which generate the highest correlation. It is also important to notice that some of these methods are not based on multi-scale processing. Thus, the input size of the images may influence their performance. In order to verify any effect regarding this issue, additional decimation steps are included at the beginning of the processing flows. These decimation steps have the function of modifying the input size of the images before computation. The performance of the methods are then analyzed over several input sizes.

Firstly, the perimeter length is discussed. In this method, the objective complexity measure is the number of pixels which belong to image edges. Here, Roberts algorithm is used to detect edges in the image. This algorithm detects edges by checking whether the gradient of pixel values is higher than a chosen threshold. The threshold value is an input setting for perimeter length and should be chosen before the method is used. Thus, it is important to analyze the behavior of the method over many threshold values. Since this method is not based on multi-scale processing, it is also interesting to analyze the effect of image size by using decimation.

In Figure E.1(a), the correlation coefficient between the objective measure and the subjective rank is given in function of the threshold value and image size. The correlation increases as the threshold value increases. It is also easy to see that reducing the size the of images before edge detection increases the correlation. The settings which generates the highest correlation ($R = 0.66$) is image size of 268 x 178 for a threshold value of 62. This is indicated in the plot by an arrow.

For the JPEG method, the objective complexity measure is the size of the digital image file. In the JPEG coding process, the input data is decomposed in several components through a transformation called *Discrete Cosine Transform*. After this decomposition, each resulting component is processed by *quantization*. Normally, this step consists of a

non-linear transformation which decreases the resolution of the amplitude values of the components.

In standard JPEG, there is parameter called quality factor Q which controls the overall loss of information during the quantization step. Factor Q can assume values in range $[1, 100]$, where higher values indicates files with higher visual quality.

For each Q, it is calculated here the correlation coefficient between the resulting JPEG file sizes and the subjective rank. Figure E.2(b) shows how the correlation depends on the parameter Q and image size. The correlation coefficient is higher for low values of quality factor. Similar to perimeter length, the correlation also increases as the image size is reduced. Interestingly, the plot shows that the effect of Q on the correlation diminishes as the image size is reduced. The JPEG settings which generates the highest correlation ($R = 0.64$) are $Q = 1$ and image size of 536 x 356 pixels. In the subband entropy, the objective measure is the Shannon entropy of wavelet coefficients. The method starts by firstly converting images into CIELab space. Then it calculates wavelets coefficients from luminance and chrominance channels in different scales. Finally, it sums the entropies of the coefficients from each scale. Figure E.2(c) shows how the correlation coefficient between this sum and the subjective rank depends on the number of scales and image size. The highest correlation ($R = 0.30$) is generated for a number of three scales (horizontal axis in Figure E.2(c)) and image size 268 x 178 pixels.

In feature congestion, the objective measure is computed as the volume of the covariance of a vector space constructed by "features" such as color, contrast and orientation. Feature congestion uses several setting parameters such as number of scales over which the covariance is computed, width of several types of filters, etc. In total, the performance of feature congestion method is analyzed here over five setting parameters. Since it is not straightforward to graphically represent a 5-D function, Figure E.2(d) exhibits only the best result found for each image size after varying the settings. Over all images sizes, 536 x 356 pixels generates the highest correlation coefficient ($R = 0.46$) for feature congestion.

Näsänen's objective measure is defined as the product between the median spatial frequency of the image energy spectrum distribution and the image area which comprises the 95% of the total image energy. Figure E.1(e) exhibits the correlation of Näsänen's measure with subjective rank in function of image size. Notice that the image area is also varied for several energy thresholds. The highest correlation is found for image size 134 x 89 pixels and area comprising 80% of the total image energy.

Finally, Figure E.1(f) gives the correlation of the proposed γ in function of image size. The highest correlation ($R = 0.72$) is found for image size of 1072 x 712 pixels.

Subjective ranks for subgroups of participants

Figure E.2 shows the subjective ranks computed considering subgroups of participants divided by nationality and gender. Table E.1 shows the correlation coefficients between the subjective ranks of the distinct subgroups.

Table E.1 shows that the perceptions annotated from different subgroups of participants are significantly correlated. This indicates that the Algerian and Japanese subjects have very similar opinions about the complexity perceived in the streetscape dataset. This is also true in case of male and female participants. However, it is clear that the influence of nationality is stronger than that of gender. Specifically, the correlation coef-

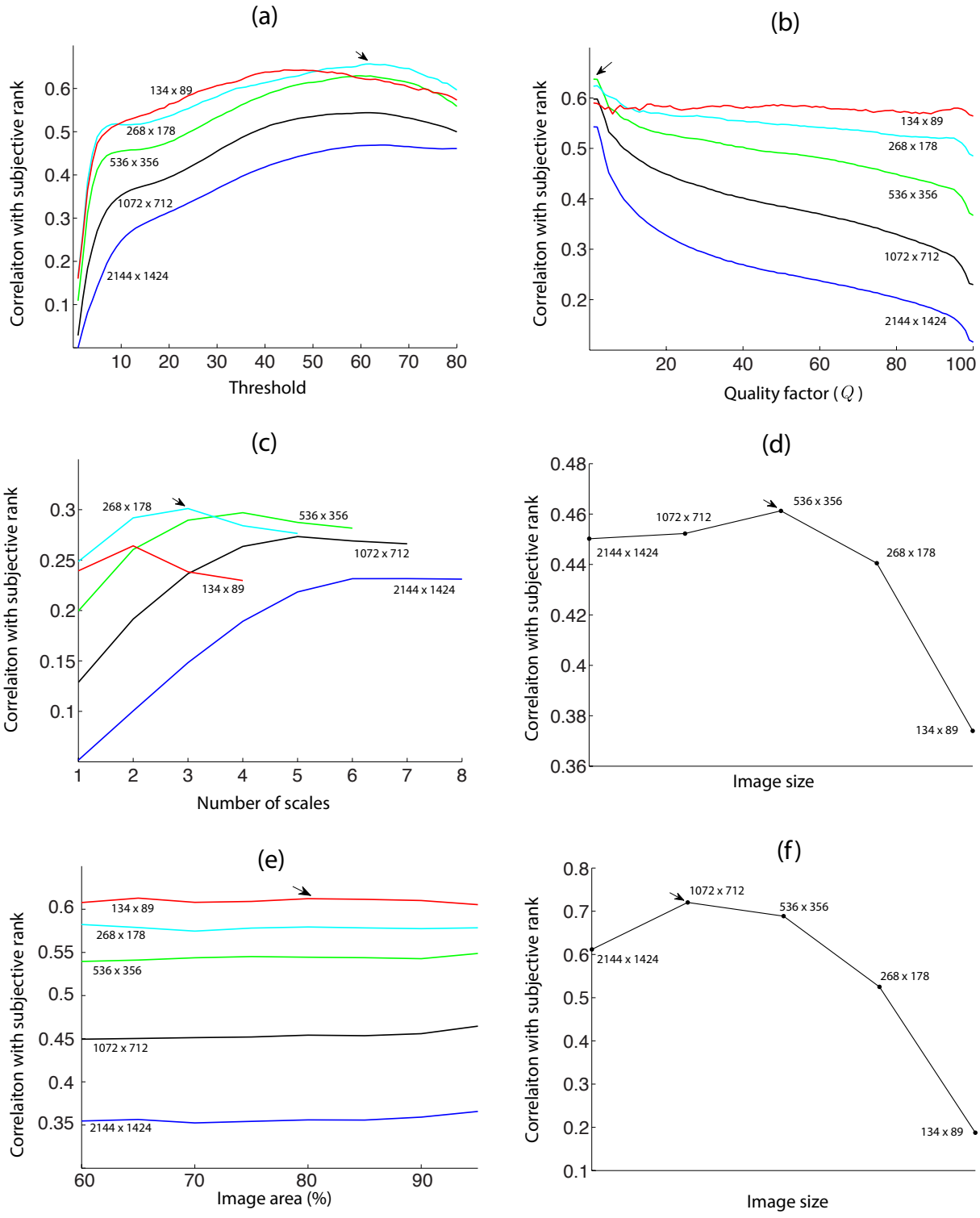

Figure E.1: **Correlation with subjective rank in function of settings**. Correlation coefficient between objective measures and the subjective rank is given in function of setting parameters. (a) Perimeter length. (b) JPEG. (c) Subband entropy. (d) Feature congestion. (e) Näsänen. (f) Proposed γ. Settings which generates the highest correlation coefficients are indicated by an arrow.

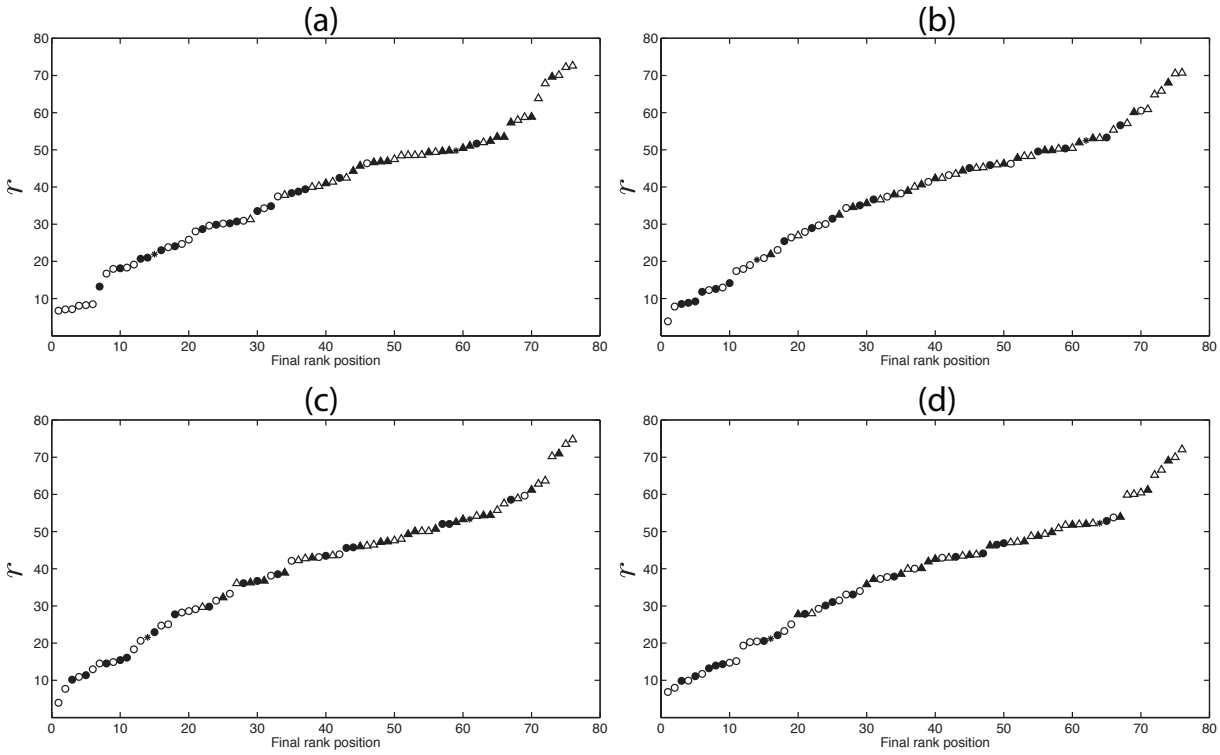

Figure E.2: **Subjective ranks generated for subgroups of participants.** (a) Algerian participants. (b) Japanese participants. (c) Males. (b) Females.

Table E.1: **Correlation coefficients between subjective ranks computed from subgroups of participants**. Values in **bold** font represents significant correlation coefficients ($p < 0.001$).

	Correlation
Algerian x Japanese	**0.87**
Males x females	**0.97**

ficient between the subjective rank from males and that from females is very high, i.e., R=0.97.

Bibliography

1. Wang D (2003) The handbook of brain theory and neural networks, MIT Press, Cambridge MA, chapter 1. 1-8. 2nd edition, pp. 1215-1219.

2. Nikhil R, Sankar K (1993) A review on image segmentation techniques. Pattern Recognition 26: 1277-1294.

3. Zucker S (1976) Region growing: childhood and adolescence. Computer graphics and image processing 5: 328-399.

4. Chen P, Pavlidis T (1980) Image segmentation as an estimation problem. Computer graphics and image processing 12: 153-172.

5. Beucher S, Meyer F (1979) Use of watersheds in contour detection. In: International workshop on image processing, real-time detection and motion detection. pp. 1-12.

6. Hoiem D, Efros A, Hebert M (2011) Recovering occlusion boundaries from an image. International journal of computer vision 91: 328-346.

7. Jain A, Duin R, Mao J (2000) Statistical pattern recognition: A review. IEEE Transactions on Pattern Analysis and Machine Intelligenc 22: 4-37.

8. Ouyang W, Tombari F, Mattoccia S, Stefano L, Cham W (2012) Performance evaluation of full search equivalent pattern matching algorithms. IEEE Transactions on Pattern Analysis and Machine Intelligence 34: 127-143.

9. Yousef M, Hussain K (2013) Fast exhaustive-search equivalent pattern matching through norm ordering. Journal of Visual Communication and Image Representation 24: 592-601.

10. Hel-Or Y, Hel-Or H (2005) Real-time pattern matching using projection kernels. IEEE Transactions on Pattern Analysis and Machine Intelligence 25: 1430-1445.

11. Taigman Y, Yang M, Ranzato M, Wolf L (2014) Deepface: Closing the gap to human-level performance in face verification. Conference on Computer Vision and Pattern Recognition (CVPR) .

12. Haralick R, Shanmugam K, Dinstein I (1973) Textural features for image classification. IEEE Transactions on Systems, Man and Cybernetics SMC-3: 610-621.

13. Tuceryan M, Jain A (1998) The Handbook of Pattern Recognition and Computer Vision, World Scientific Publishing Co., chapter Texture Analysis. pp. 207-248.

14. Koch C, Ullman S (1985) Shifts in selective visual attention: towards the underlying neural circuitry. Human Neurobiology 4: 219-227.

15. Itti L, Koch C (2000) A saliency-based search mechanism for overt and covert shifts of visual attention. Vision Research 40: 1489-1506.

16. Hansen B, Hess R (2007) Structural sparseness and spatial phase alignment in natural scenes. Journal of the Optical Society of America A 24: 1873-1885.

17. Liu Z, Li W, Shen L, han Z, Zhang Z (2010) Automatic segmentation of focused objects from images with low depth of field. Pattern Recognition Letters 31: 572-581.

18. Forster B, VanDeVille D, Berent J, Sage D, Unser M (2004) Complex wavelets for extended depth-of-field: a new method for the fusion of multichannel microscopy images. Microscopy Research and Technique 65: 33-42.

19. Wu Q, Merchant F, Castleman K (2010) Microscope image processing. Academic Press; 1st edition.

20. Harguess J, Larson J (2013) Vegetation versus man-made object detection from imagery for unmanned vehicles in off-road environments. Proc SPIE Unmanned Systems Technology XV 8741.

21. Bradley D, Unnikrishnan R, Bagnell J (2007) Vegetation detection for driving in complex environments. In: IEEE International Conference on Robotics and Automation. pp. 1-8.

22. Sofman B, Bagnell J, Stentz A, Vandapel N (2006) Terrain classification from aerial data to support ground vehicle navigation. Technical Report CMU-RI-TR-05-39, The Robotics Institute of Carnegie Mellon University.

23. Hinz S, Baumgartner A (2003) Automatic extraction of urban road networks from multi-view aerial imagery. ISPRS Journal of Photogrammetry and Remote Sensing 58: 83-98.

24. Youn J, Bethel J, Mikhail E, Lee C (2008) Extracting urban road networks from high-resolution true orthoimage and lidar. Photogrammetric Engineering & Remote Sensing 74: 227-237.

25. Tiede D, Lang S, Fureder P, Holbling D, Hoffmann C, et al. (2011) Automated damage indication for rapid geospatial reporting. Photogrammetric Engineering & Remote Sensing 77: 933-942.

26. Turker M, Sumer E (2008) Building based damage detection due to earthquake using the watershed segmentation of the post event aerial images. International Journal of Remote Sensing 29: 3073-3089.

27. Peijun L, Haiqing X, Jiancong G (2010) Urban building damage detection from very high resolution imagery using ocsvm and spatial features. International Journal of Remote Sensing 31: 3393-3409.

28. Carlotto M (1997) Detection and analysis of change in remotely sensed imagery with application to wide area surveillance. IEEE Transactions on Image Processing 6: 189-202.

29. Kessel R (2003) Texture-based discrimination of man-made and natural objects in sidescan sonar imagery. Signal Processing, Sensor Fusion, and Target Recognition XII, Proceedings of SPIE 5096: 160-168.

30. Fan H (2014) Identifying man-made objects along urban road corridors from mobile lidar data. IEEE Geoscience and Remote Sensing Letters 11: 950-954.

31. Carlotto M (2000) Nonlinear background estimation and change detection for wide area search. Optics Engineering 39: 1223-1229.

32. Cai F (2011) Man-made object detection based on texture clustering and geometric structure feature extracting. IJ Information Technology and Computer Science 2: 9-16.

33. Carlotto M (2005) A cluster-based approach for detecting man-made objects and changes in imagery. IEEE Transactions on Geoscience and Remote Sensing 43: 374-387.

34. Rapoport AB (1987) Pedestrian street use: Culture and perception. In: Anne Vernez Moudon (ed.), New York: Columbia University Press. pp. 80-92.

35. Ewing R, Handy S (2009) Measuring the unmeasurable: urban design qualities related to walkability. Journal of Urban Design 14: 65-68.

36. Ewing R, Handy S, Brownson R, Clemente O, Winston E (2006) Identifying and measuring urban design qualities related to walkability. Journal of Physical Activity and Health 3: 223-240.

37. Wolfe J (2011) What are the shapes of response time distributions in visual search? Journal of Experimental Psychology 37: 58-71.

38. Silva MD, Courboulay V, Estraillier P (2011) Image complexity measure based on visual attention. In: 18th IEEE International Conference on Image Processing (ICIP). pp. 1-8.

39. Rosenholtz R, Li Y, Nakano L (2007) Measuring visual clutter. Journal of Vision 7: 1-22.

40. Wolfe J, Alvarez G, Rosenholtz R, Kuzmova Y, Sherman A (2011) Visual search for arbitrary objects in real scenes. Atten Percept Psychophys 73: 1650-1671.

41. Caroux L, Bigot L, Vibert N (2014) Impact of the motion and visual complexity of the background on players's performance in video game-like displays. Ergonomics 56: 1863-1876.

42. Fritsch J, Kuehnl T, Geiger A (2013) A new performance measure and evaluation benchmark for road detection algorithms. In: International Conference on Intelligent Transportation Systems (ITSC). pp. 1-8.

43. Marciano H, Yeshurun Y (2014) Perceptual load in different regions of the visual scene and its relevance for driving. Human factors in press.

44. Jahn G, Oehme A, Krems J, Gelau C (2005) Peripheral detection as a workload measure in driving: Effects of traffic complexity and route guidance system use in a driving study. Transportation Research Part F 8 : 255-275.

45. Edquist J, Rudin-Brown C, Lenné M (2011) Speed choice and hazard perception in complex urban road environments with and without on-street parking. In: Australian Road Safety Research, Education & Policing Conference. pp. 1-8.

46. Edquist J, Rudin-Brown C, Lenné M (2012) The effects of on-street parking and road environment visual complexity on travel speed and reaction time. Accident Analysis and Prevention : 759-765.

47. C M Rudin-Brown ML J Edquist (2014) Effects of driving experience and sensation-seeking on drivers' adaptation to road environment complexity. Safety Science : 121-129.

48. Fildes B, Lee S (1993) The speed review: Road environment, behaviour, speed limits, enforcement and crashes. Technical Report CR 127 (FORS); CR 3/93 (RSB), MUARC, for Federal Office of Road Safety (FORS) and Road Safety Bureau, Roads and Traffic Authority NSW (RSB).

49. Kaplan S, Kaplan R, Wendt J (1972) Rated prefererence and complexity for natural and urban visual material. Perception and Psychophysics 12: 354-356.

50. R Ewing KB (2013) Pedestrian- and Transit-Oriented Design. Urban Land Institute.

51. Hyvärinen A, Karhunen J, Oja E (2001) Independent Component Analysis. John Wiley and Sons, Inc.

52. Dyson F (1943) A note on kurtosis. Journal of the Royal Statistical Society 106: 360-361.

53. Finucan H (1964) A note on kurtosis. Journal of the Royal Statistical Society, Series B 26: 111-112.

54. Balanda K, MacGillivray H (1988) Kurtosis: A critical review. The American Statistician 42: 111-119.

55. DeCarlo L (1997) On the meaning and use of kurtosis. Psychological Methods 2: 292-307.

56. Tsai D, Wang H (1998) Segmenting focused obects in complex visual images. Pattern Recognition Letters 19: 929-940.

57. Wang J, Li J, Gray R, Wiederhold G (2001) Unsupervised multi resolution segmentation for images with low depth of field. IEEE Transactions on Pattern Analysis and Machine Intelligenc 23: 85-90.

58. Kim C (2005) Segmenting a low-depth-of-field image using segmenting a low-depth-of-field image using morphological filters and region merging. IEEE Transactions on Image Processing 14: 1503-1511.

59. Li H, Ngan K (2007) Unsupervized video segmentation with low depth of field. IEEE Transactions on circuits systems and video technology 17: 1742-1751.

60. Zhang K, Lu H, Wang Z, Zhao Q, Duan M (2006) A fuzzy segmentation of salient region of interest in low depth of field image. In: Proc. 13th International Conference on Multimedia Modeling. pp. 782-791.

61. Graf F, Kriegel H, Weiler M (2011) Robust segmentation of relevant regions in low depth of field images. In: 18th IEEE International Conference on Image Processing (IEEE ICIP2011). pp. 2861 - 2864.

62. Mei J, Si Y, Gao H (2013) A curve evolution approach for unsupervised segmentation of images with low depth of field. IEEE Transactions on Image Processing 22: 4086-4095.

63. Chan T, Vese L (2001) Active contours without edges. IEEE Transactions on Image Processing 10: 266-277.

64. Cheng M, Warrell J, Lin WY, Zheng S (2013) Efficient salient region detection with soft image abstraction. In: IEEE International Conference on Computer Vision. pp. 1529-1536.

65. Rahtu E, Kannala J, Salo M, Heikkilä J (2010) Segmenting salient objects from images and videos. In: Computer Vision – ECCV 2010. volume 6315, pp. 366-379.

66. Achanta R (2010) Saliency detection using maximum symmetric surround. In: International Conference on Image Processing. pp. 1-8.

67. Otsu N (1979) A threshold selection method from gray-level histograms. IEEE Transactions on Systems, Man and Cybernetics 9: 62-66.

68. Olmos A, Kingdom F (2004) A biologically inspired algorithm for the recovery of shading and reflectance images. Perception 33: 1463-1473.

69. Hyvärinen A (1999) Fast and robust fixed-point algorithms for independent component analysis. IEEE Transactions on Neural Networks 10: 626-634.

70. Cavalcante A, Barros A, Takeuchi Y, Ohnishi N (2011) Effects of second-order statistics on independent component filters. In: ICONIP, LNCS. volume 7062, pp. 54-61.

71. Cavalcante A, Lucena F, Barros A, Takeuchi Y, Ohnishi N (2010) Analyzing differences between gabor functions and ica filters learned from natural scenes. Australian Journal of Intelligent Information Processing Systems 12: 41-45.

72. Martin D, Fowlkes C, Tal D, Malik J (2001) A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: Proc. 8th IEEE International Conference Computer Vision. volume 2, pp. 416-423.

73. Cavalcante A, Barros A, Takeuchi Y, Ohnishi N (2012) Segmentation of depth-of-field images based on the response of ica filters. IEICE TRANSACTIONS on Information and Systems E95-D: 1170-1173.

74. Anderson H (1987) Edge detection for object recognition in aerial photographs. Technical report, University of Pennsylvania.

75. Cleynenbreugel J, Suetens P, Fierens F, Wambacq P, Oosterlinck A (1989) A knowledge-based system for the recognition of roads on spot satellite image. In: SPIE Advance in Image Compression and automatic target recogntion. volume 1099, pp. 83-88.

76. Bhagavathy S, Manjunath B (2006) Modeling and detection of geospatial objects using texture motifs. IEEE Transactions on Geoscience and Remote Sensing 44: 3706-3715.

77. Porway J, Wang K, Yao B, Zhu SC (2008) A hierarchical and contextual model for aerial image understanding. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 1-8.

78. Yang L, Wu X, Praun E, Ma X (2009) Tree detection from aerial imagery. In: Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems. pp. 131-137.

79. Torralba A, Oliva A (2003) Statistics of natural image categories. Network: Computation in Neural Systems 14: 391-412.

80. Caron Y, Makris P, Vicent N (2002) A method for detecting artificial objects in natural environments. In: 16th International Conference on Pattern Recognition. Proceedings. volume 1, pp. 600-603.

81. Kumar S, Hebert M (2003) Man-made structure detection in natural images using a causal multiscale random field. In: Computer Vision and Pattern Recognition. Proceedings. IEEE International Conference on Computer Vision and Pattern Recognition. volume 1, pp. 119-126.

82. Kumar S, Hebert M (2004) Discriminative fields for modeling spatial dependencies in natural images. In: Advances in Neural Information Processing Systems. volume 16, pp. 1-8.

83. Vishwanathan S, Schraudolph N, Schmidt M, Murphy K (2006) Accelerated training of conditional random fields with stochastic gradient methods. In: Proceedings of the 23rd International Conference on Machine Learning. pp. 969-976.

84. Cavalcante A, Lucena F, Barros A, Takeuchi Y, Ohnishi N (2009) Segmentation of natural and man-made structures by independent component analysis. In: Independent Component Analysis and Signal Separation. volume 5441, pp. 483-490.

85. Coughlan J, Yuille A (2003) Manhattan world: Orientation and outlier detection by bayesian inference. Neural Computation 15: 1063 - 1088.

86. Denis P, Elder J, Estrada F (2008) Efficient edge-based methods for estimating manhanttan frames in urban imagery. In: Proc. European Conference on Computer Vision (5303). pp. 197-211.

87. Majdik A, Albers-Schoenberg Y, Scaramuzza D (2013) Mav urban localization from google street view data. In: IEEE/RSJ International Conference on Intelligent Robots and Systems. pp. 3979-3986.

88. Attneave F (1957) Physical determinants of the judged complexity of shapes. Journal of Experimental Psychology 53: 221-227.

89. Campbell F, Robson J (1968) Application of fourier analysis to the visibility of gratings. Journal of Physiology 197.

90. Ginsburg A (1971) Psychological correlates of a model of the human visual system. IEEE Transaction on Aerospace and Electronic Systems 71-C-AES.

91. Ginsburg A (1986) Spatial filtering and visual form perception. Handbook of Perception and Human Performance 34.

92. Piotrowski L, Campbell F (1981) A demonstration of the visual importance and flexibility of spatial-frequency amplitude and phase. Perception 11: 337-346.

93. Näsänen R, Kukkonen H, Rovamo J (1993) Spatial integration of band-pass filtered patterns in noise. Vision Research 33: 903-911.

94. Chikhman V, Bondarko V, Danilova M, Goluzina A, Shelepin Y (2012) Complexity of images: experimental and computational estimates compared. Perception 41: 631-647.

95. Forsythe A, Sheehy N, Sawey M (2003) Measuring icon complexity: An automated analysis. Behavior Research Methods 35: 334-342.

96. Rosenholtz R, Li Y, Mansfield J, Jin Z (2005) Feature congestion, a measure of display clutter. ACM Special Interest Group on Computer Human Interaction : 761-770.

97. Donderi D (2006) An information theory analysis of visual complexity and dissimilarity. Perception 35: 823-835.

98. Forsythe A, Nadal M, Sheehy N, Cela-Conde C, Sawey M (2011) Predicting beauty: Fractal dimension and visual complexity in art. British Journal of Psychological 102: 49-70.

99. Elsheshtawy Y (1997) Urban complexity: Toward the measurements of the physical complexity of street-scapes. Journal of Architectural and Planning Research 14: 301-316.

100. Cooper J (2003) Fractal assessment of street-level skylines: a possible means of assessing and comparing character. Urban Morphology 7: 73-82.

101. Mansouri L, Matsumoto N, Cavalcante A, Mansouri A (2013) Study on subjective visual complexity and rms image contrast statistics in streetscapes in algerian and japan. AIJ Journal of Architecture Planning 78: 625-633.

102. Cavalcante A, Mansouri A, Kacha L, Barros A, Takeuchi Y, et al. (2014) Measuring streetscape complexity based on the statistics of local contrast and spatial frequency. PLoS ONE 9: e87097.

103. Peli E (1990) Contrast in complex images. Journal of the Optical Society of America A 7: 2032-2040.

104. Delplanque S, N'diaye K, Scherer K, Grandjean D (2007) Spatial frequencies or emotional effects? a systematic measure of spatial frequencies for iaps pictures by a discrete wavelet analysis. Journal of Neuroscience Methods : 144-150.

105. Doshi A, Trivedi M (2012) Head and eye gaze dynamics during visual attention shifts in complex environments. Journal of Vision 12: 1-16.

106. Murray S, Olshausen B, Woods D (2003) Processing shape, motion, and three-dimensional shape-from-motion in the human cortex. Cerebral Cortex 13: 508-516.

107. Frazor RA, Geisler WS (2006) Local luminance and contrast in natural images. Vision Research 46: 1585-1598.

108. Movshon JA, Thompson ID, Tolhurst DJ (1978) Spatial summation in the receptive field of simple cells in the cat's striate cortex. Journal of Physiology (London) 283: 53-77.

109. Dean AF (1981) The relationship between the response amplitude and contrast for cat striate cortical neurones. Journal of Physiology (London) 318: 413-427.

110. Albrecht DG, Hamilton DB (1982) Striate cortex of monkey and cat: contrast response function. Journal of Neurophysiology 43.

111. Bex PJ, Solomon SG, Dakin SC (2009) Contrast sensitivity in natural scenes depends on edge as well as spatial frequency structure. Journal of Vision 9: 1-19.

112. Hecht S (1928) The relation between visual acuity and illumination. Journal of General Physiology 11: 255-281.

113. Campbell FW, Gubish RW (1966) Optical quality of human eye. Journal of Physiology (London) 186: 558-578.

114. Lindh U (2012) Light Shapes Spaces: Experience of Distribution of Light and Visual Spatial Boundaries. Ph.D. thesis, HDK - School of Design and Crafts, University of Gothenburg.

115. Hubel DH, Wiesel TN (1962) Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. Journal of Physiology (London) 160: 106-154.

116. Blakemore C, Campbell FW (1969) On the existence of neurones in the human visual system selectively sensitive to the orientation and size of retinal images. Journal of Physiology 203.

117. Tolhurst DJ (1972) On the possible existance of edge detector neurones in the human visual system. Vision Research 5.

118. Shapley RM, Tolhurst DJ (1973) Edge detectors in human vision. Journal of Physiology 229: 165-183.

119. Olshausen BA, Field DJ (1996) Emergence of simple-cell receptive field properties by learning a sparse code for natural images. Nature 381: 607-609.

120. Field D (1994) What is goal of sensory coding. Neural Computation 6: 559-601.

121. S Plainis IM (2002) Reaction times as an index of visual conspicuity when driving at night. Ophthalmic Physiol Opt : 409-415.

122. Chandler D, Field D (2007) Estimates of the information content and dimensionality of natural scenes from proximity distributions. Journal of Optic Society of America A 24: 922-941.

123. Yeragani V, Srinivasan K, Vempati S, Pohl R, Balon R (1993) Fractal dimension of heart rate time series: an effective measure of autonomic function. Journal of Applied Physiology 75: 2429-2438.

124. Lehky S, Sejnowski T, Desimone R (2005) Selectivity and sparseness in the responses of striate complex cells. Vision Research 45: 57-73.

125. Olshausen B, Field D (2005) How close are we to understanding v1? Neural Computation 17: 1665-1699.

126. Luenberger D (1969) Optimization by Vector Space Methods. New York Wiley.

127. Hyvärinen A, Hoyer P, Inki M (2001) Topograpgic independent component analysis. Neural Computation 13: 1527-1558.

128. Sharpee T, Sugihara H, Kurgansky A, Rebrik S, Stryker M, et al. (2006) Adaptive filtering enhances information transmission in visual cortex. Nature 439: 936942.