UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO

ALI KARAALI

Spatially Varying Defocus Blur Estimation and Applications

Dissertation presented in partial fulfillment
of the requirements for the degree of
Doctor of Computer Science

Advisor: Prof. Dr. Cláudio Rosito Jung

Porto Alegre
April 2017

CIP — CATALOGING-IN-PUBLICATION

Karaali, Ali

Spatially Varying Defocus Blur Estimation and Applications / Ali Karaali. – Porto Alegre: PPGC da UFRGS, 2017.

121 f.: il.

Dissertation (Ph.D.) – Universidade Federal do Rio Grande do Sul. Programa de Pós-Graduação em Computação, Porto Alegre, BR–RS, 2017. Advisor: Cláudio Rosito Jung.

1. Spatially Varying Defocus. 2. Defocus Blur Estimation. 3. Image retargeting. 4. Image Deblurring. 5. Video Deblurring. I. Rosito Jung, Cláudio. II. Título.

"As much as we may like some things to remain suspended in time,

they never do."

— Dexter Morgan

ACKNOWLEDGMENTS

Competing PhD! It was one of the most challenging activities in my life including passing the Celpe-Bras Portugues exam. First of all, I would like to thank to my advisor, Prof. Dr. Cláudio Rosito Jung for his guidance, patient, support and accepting me as a PhD student. I really felt that this dissertation would not arrive this point without his guidance.

I would like to thank to my dissertation committee, Prof. Dr. Manuel Oliveira, Prof. Dr. Luiz Gonzaga da Silva and Prof. Dr. Eduardo Silva for their guidance and helpful suggestions. Many thanks to Prof. Dr. Marcelo Walter for his support too.

Ve öncelikle babam Mustafa Karaali olmak üzere tüm aile fertlerime, annem Cemile Karaali'ye, ağabeyim Aydın Karaali ve ailesine teşekkürü bir borç bilirim.

Finalmente, eu também gostaria de agradecer a todas as pessoas da sala 67-232 pelas nossas conversas acadêmicas e off-topics.

ABSTRACT

This dissertation presents two different defocus blur estimation methods for still images. Both methods assume a Gaussian Point Spread Function (PSF) and explore the ratio of gradient magnitudes of reblurred images computed at edge location with different scales, which provides a closed form mathematical formulation for the local blur assuming continuous-time signals. The first approach computes 1D profiles along edge points orthogonal to the local contour, and evaluate the location of the edge (maximum of the derivative) to adaptively select the number of reblurring scales. Considering the time consumption of exploring 1D oriented edge profiles, a second method was proposed based on 2D multiscale image gradients, and local reblurring parameters were selected based on the agreement of an edge detector computed at several scales.

Given an initial estimate of the blur scale at edge locations provided by either of these two methods, a correction step that accounts for the discretization of the continuous formulation is also proposed. A novel local filtering method that smooths the refined estimates along the image contours is also proposed, and a fast joint domain filter is explored to propagate blur information to the whole image to generate the full blur map. Experimental results on synthetic and real images show that the proposed methods have promising results for defocus blur estimation, with a good trade off between running time and accuracy when compared to state-of-the art defocus blur estimation methods.

To deal with blurry video sequences, temporal consistency was also included in the proposed model. More precisely, Kalman Filters were applied to generate smooth temporal estimates for each pixel when the local appearance of the video sequence does not vary much, and allowing sharp transitions during drastic local appearance changes, which might relate to occlusions/disocclusions.

Finally, this dissertation also shows applications of the proposed methods for image and video blur estimation. A new image retargeting method is proposed for photos taken by a shallow Depth of Field (DoF) camera. The method includes defocus blur information with the seam carving framework aiming to preserve in-focus objects with better visual quality. Assuming the in-focus pixels related to regions of interest of a blurry image, the proposed retargeting method starts with a cropping method, which removes the unimportant parts (blurry) of the image, then the seam carving

method is applied with a novel energy function that prioritizes in-focus regions. Experimental results show that the proposed blur aware retargeting method works better at preserving in-focus objects than other well known competitive retargeting methods.

The dissertation also explores the proposed blur estimation method in the context of image and video deblurring, and results were compared with several other blur estimation methods. The obtained results show that metrics typically used to evaluate blur estimation methods (e.g. Mean Absolute Error) might not be correlated with the quality of deblurred image metrics, such as Peak Signal to Noise Ratio.

Keywords: Spatially Varying Defocus. Defocus Blur Estimation. Image retargeting. Image Deblurring. Video Deblurring.

Estimação de Borramento por Desfoco Espacialmente Variante e Aplicações

RESUMO

Esta tese apresenta dois métodos diferentes de estimativa de desfocagem usando uma única imagem. Ambos os métodos assumem uma função de espalhamento de ponto (*Point Spread Function* - PSF) Gaussiana e exploram a razão de magnitudes de gradientes de versões re-borradas da imagem original com escalas diferentes nas bordas da imagem, o que fornece uma expressão matemática fechada para borramento local. A primeira abordagem calcula perfis 1D ao longo de pontos de borda ortogonais ao contorno local, e avalia a localização da borda (máximo da derivada primeira) para selecionar adaptativamente o número de escalas no re-borramento. Considerando o consumo de tempo de explorar perfis de aresta orientados 1D, um segundo método foi proposto com base em gradientes de imagem diretamente no domínio 2D, e os parâmetros de re-borramento locais foram selecionados com base na concordância de um detector de bordas calculado em várias escalas.

Dada uma estimativa inicial da escala de desfocagem nas posições de borda proporcionada por qualquer um destes dois métodos, é também proposto um passo de correção que atenua os erros introduzidos pela discretização da formulação contínua. Um novo método de filtragem local que suaviza as estimativas refinadas ao longo dos contornos de imagem também é proposto, e um filtro de domínio conjunto (*joint-domain filter*) rápido é explorado para propagar informações de desfocagem para toda a imagem, gerando o mapa de desfocagem completo. Os resultados experimentais em imagens sintéticas e reais mostram que os métodos propostos apresentam resultados promissores para a estimativa de borramento por desfoco, com um bom compromisso entre qualidade e tempo de execução quando comparados a técnicas estado-da-arte.

Para lidar com sequências de vídeo desfocadas, a consistência temporal também foi incluída no modelo proposto. Mais precisamente, Filtros de Kalman foram aplicados para gerar estimativas temporais suaves para cada pixel quando a aparência local da sequência de vídeo não varia muito, permitindo transições durante mudanças drásticas da aparência local, que podem se relacionar com oclusões/desoclusões.

Finalmente, esta tese também mostra aplicações dos métodos propostos para a esti-

mativa de desfocagem de imagem e vídeo. Um novo método de redimensionamento (*retargeting*) de imagens é proposto para fotos tiradas por câmera com baixa profundidade de campo. O método inclui informação de desfocamento local no contexto do método *seam carving*, visando preservar objetos em foco com melhor qualidade visual. Assumindo que os pixels em foco estejam relacionados às regiões de interesse de uma imagem com desfocamento, o método de redimensionamento proposto começa com um método de corte (*cropping*), o qual remove as partes sem importância (borradas) da imagem, e então o método *seam carving* é aplicado com uma nova função de energia que prioriza as regiões em foco. Os resultados experimentais mostram que o método proposto funciona melhor na preservação de objetos em foco do que outras técnicas de redimensionamento de imagens.

A tese também explora o método de estimação de desfocagem proposto no contexto de des-borramento de imagens e sequências de vídeo, e os resultados foram comparados com vários outros métodos de estimação de desfocagem. Os resultados obtidos mostram que as métricas tipicamente usadas para avaliar métodos de estimação de desfocagem (por exemplo, erro absoluto médio) podem não estar correlacionadas com a qualidade das métricas de imagem desfocada, como a Relação Sinal-Ruído de Pico.

Palavras-chave: Estimação de borramento, Remapeamento de imagens, Desborramento.

LIST OF ABBREVIATIONS AND ACRONYMS

CEF	Connected Edge Filter
CoC	Circle of Confusion
ISC	Improved Seam Carving
JNB	Just Noticeable Blur
LSF	Least Square Fitting
LBP	Local Binary Pattern
MAE	Mean Absolute Error
MSE	Mean Square Error
OSS	Optimized Scale and Stretch
PSNR	Peak Signal to Noise Ratio
PSF	Point Spread Function
SDD	Symmetric Diagonally-Dominant
STD	Structure-Texture Decomposition
WARP	Nonhomogeneous Warping

LIST OF FIGURES

LIST OF TABLES

CONTENTS

1 INTRODUCTION

Images are acquired to record or show useful information in many areas, such as consumer photography, hyper-spectral imaging, microscopy, medical imaging, etc. However, all these imaging systems have imperfections in their sensors and/or circuitry of the cameras. Hence, most of the recorded images represent some amount of degradation of the real scene depending on camera settings and lighting conditions, among other factors. One of the degradation types that can occur is blur, which arises due to several factors.

1.1 Sources of Blur

Considering the blur sources, one can categorize the blur mainly into two groups: defocus and motion blur (see examples in Fig. 1.1).

Figure 1.1: (a) Motion and (b) defocus blur effect on images. Source: <http://en.wikipedia.org/wiki/Blur> accessed in January, 2017

(a) (b)

Imaging systems create photos assuming that the sensor will capture only a single instant of time. However, because of hardware constraints (or some artistic reasons), the camera senses the scene over a period of time. Mostly, this exposure period is short enough to capture an instantaneous moment of the scene, but this is not always true. Due to the camera or object movement, which is amplified by long exposure times, the image may present artifacts due to the motion and this artifact is called motion blur (see example in Fig. 1.1(a)).

On the other hand, limited focal range of optical lenses results in defocus blur, which is a phenomenon that can be encountered on a daily basis. Defocus blur is often called as a defect, but differently from motion blur it is not just an unwanted

inevitable degradation of an imaging system. In fact, defocus blur can be used on purpose in photographs in order to increase the visual importance of a subject, to attenuate the viewers' attention from "unimportant" parts of the photograph, or just because of artistic reasons. This phenomenon can be observed in Fig. 1.2. The object of interest, which is the top of the fountain, can be immediately perceived as the most relevant part of the image. In fact, this object is depicted in focus (sharply), whereas all the other parts of the photograph are blurred.

Figure 1.2: A photograph as an example for DOF in photography. Source: <http://androidnewspad.com/tag/battle/> accessed in January, 2017

Apart from the diversity of occurrence of defocus blur on photographs, estimating defocus blur is a challenging task, mainly because the blur amount is spatially varying and cannot be represented by a unique global descriptor. Most of the existing methods for defocus blur identification estimate the *Point Spread Function* (PSF) associated with the image acquisition system, where the PSF typically is spatially varying and basically described through a simple model (e.g. disc, Gaussian), which is represented by a single parameter that shows its scale (radius, standard deviation etc.). For a given image, a 2D map of the scale parameter is called ***defocus blur map***, which illustrates the local defocus blur amount at each pixel location (see an example in Fig. 1.3).

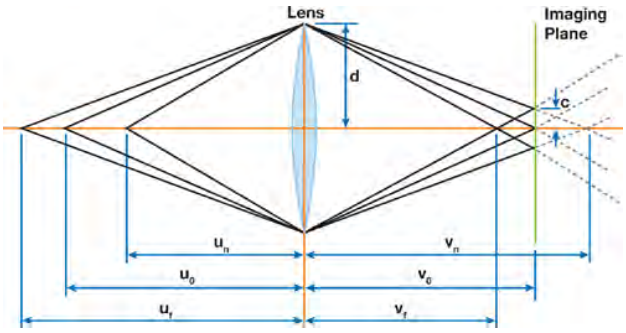
1.2 Degradation Model of the Blur

Even though there are various kinds of blur types (as observed in Fig. 1.1), blur can be modeled by similar degradation model. However, this dissertation will tackle the defocus blur degradation model. A brief explanation of this model can be as following: from a given camera setting, rays coming from scene points that lie on

Figure 1.3: Original and its defocus blur map image (produces by the method of (ZHUO; SIM, 2011))

the focal plane, which is located at a certain distance from the lens, will converge exactly at the same point. However, rays coming from points that lie on farther or closer distances away from the focal plane will not converge to the same point, but to a patch on the sensor plane, called Circle of Confusion (CoC) (as can be seen in Fig. 1.4). This latter regions are referred to as defocused (PENTLAND, 1987).

Figure 1.4: Circle of Confusion in thin lens model Source: <http://http.developer.nvidia.com/GPUGems3/gpugems3_ch28.html> accessed January, 2017



A defocus image can be formulated as convolution of the image and defocus function Point Spread Function (PSF) plus noise. Considering a spatially invariant PSF, the model can be formulated as

$$I_b(x, y) = I(x, y) * h(x, y) + \eta(x, y), \tag{1.1}$$

where $I_b(x, y)$ and $I(x, y)$ represent the observed and the original images respectively, $h(x, y)$ is the PSF of blur and η is the noise parameter. In this model, h is the defocus kernel, typically a Gaussian function with spread parameter σ_b, which is related to the amount of blur. Therefore, the quality of blur parametrization is highly related to the defocus function (or Point Spread Function).

The model given in Eq. 1.1 represents spatially constant blur, meaning that

each pixel has the same blur scale. However, as it can be observed in Fig. 1.4, the spread parameter σ_b is related to the distance of the object. Therefore, different regions should have different spread parameters. A more realistic formulation of defocus blur is given by

$$I_b(x, y) = I(x, y) * h(x, y; \sigma(x, y)) + \eta(x, y), \tag{1.2}$$

where $h(x, y; \sigma(x, y))$ indexed PSF with a spatially varying scale parameter $\sigma(x, y)$ and η is the noise parameter.

Using a spatially varying Gaussian kernel to model blur degradation is more realistic than to assume the blur kernel is fixed for whole image pixels. However, the difference in spatially varying case is high and any kind of measurement strategy needs some local assumptions, otherwise its model can be extremely complicated.

1.3 Motivation for This Work

Even though blurring causes the loss of image details and is often undesired, estimating the spatially varying blur amount from a single image might provide several additional cues about the scene captured by the camera. In particular, defocus blur estimation has been studied by researchers in computer vision, computational photography and computer graphics for variety of applications, such as

- **Deblurring** : it can be referred as the main application of the defocus blur and it is basically retrieving back the sharpness of the blurry regions to generate a partially or an all-focus image.

- **Refocusing** : it is a research topic related to computational photography and basically is an image processing technique that magnifies existing defocus from a given single photo in order to divert the importance of the object in the scene.

- **Retargeting** : it a is relatively recent research area and basically related to changing the size of the image for desired display that might include changing the aspect ratio of the image while keeping the important objects with an acceptable visual quality. Retargeting is proposed because the classical image resizing operators (e.g. scaling, cropping) are inefficient to keep the content

of an image, since these methods tent to cause distortions and/or remove important information from the image. This dissertation presents a blur-aware retargeting approach.

The aforementioned applications for defocus blur estimation show that it is an important problem in vision and graphics. Although there are many existing approaches for defocus blur estimation, the problem still presents many challenges (some of these methods and challenges will be discussed in Chapter 2), particularly in the context of blurry video sequences. In this dissertation we propose an edge-based defocus blur estimation method for still images and also an extension for video sequences aiming to remove temporal flickering. We also show applications of the proposed blur estimation method, such as image and video deblurring, and introduce a new blur-aware retargeting method.

1.4 Goals

The main goal of this dissertation is to develop an edge-based defocus blur estimation method that tackles the most common problems for any edge-based defocus blur map estimation methods might face: noise, edge mislocalization and edge interference. Although most of the existing edge-based methods use gradient magnitude ratios of the original and/or reblurred versions of the image, there is no discussion about adequate reblurring parameter(s) selection, which is a key issue for any kind of edge-based defocus blur estimation methods. In this dissertation, we aim to explore the selection of reblurring parameter for gradient magnitude calculation to cope the aforementioned problems. Moreover, we aim to propose the use of a fast guided filter to propagate estimated blur amounts from edge location to whole image.

Then, we want to extend the proposed method for video-related applications. Since the use of defocus blur estimation methods in a frame by frame manner tends to produce temporal artifacts, another goal of this dissertation is to enforce some temporal coherence into the proposed framework. Finally, we aim to explore the estimated defocus blur maps in different applications, such as image/video deblurring, and blur-aware image retargeting.

2 RELATED WORK

The challenging problem of defocus blur estimation from a single image has been studied before and several methods have been proposed in the past years. Nevertheless, it is a long-standing problem since measuring the defocus level of each pixel is not straightforward due to a variety of factors, such as noise, lack of information in homogeneous regions, etc. Therefore, it is still an active research area.

In (VU; PHAN; CHANDLER, 2012), the authors classify image blur/sharpness estimation into three classes: edge-based, pixel-based and transform-based methods. Since both pixel-based and transform-based methods explore local image patches, in this dissertation we consider two main classes of defocus blur estimation methods: edge-based and patch-based. The first class initially models and estimates defocus blur along image edges (where blur is more noticeable), and then propagates the information through the whole image. The second class estimates the full blur map directly at all pixels, using mostly local patches/regions. Some defocus blur estimation algorithms will be revised next.

2.1 Edge-Based Methods

One of the earliest attempts for estimating the defocus amount on images is the work of (PENTLAND, 1987). In this method, a correlation was obtained between the degree of blur and the depth by separating the characteristic of the scene from the lens via analyzing the effect of sharp edges in the image data. The method represents a blurry edge as a convolution of the image data with a Gaussian (PSF), which is defined by $h(r; \sigma_b)$, where r is the blur radius and σ_b is the measure of spread (standard deviation). The author explored the Laplacian of this convolution, given by

$$C(x, y) = \nabla^2(h(r; \sigma_b) * I(x, y)), \tag{2.1}$$

and used it at the point of zero crossings in order to estimate σ_b. One limitation of this method is feature dependency, since the scene characteristic must be known prior to evaluation. In the paper, the results are given qualitatively according to the human visual perception.

Later, (ELDER; ZUCKER, 1998) described a simultaneous edge detection and blur estimation method, that exploits the first and second order derivatives, obtained using steerable Gaussian basis filters. In particular, they compute the distance between extrema of inverse signs of the second derivative in the gradient direction using steerable filters as depicted in Fig. 2.1. Then, these second order derivatives are fitted to a step edge model in order estimate the exact location of the blurry edge, and the distance parameter d is used to compute the unknown blur amount of the blurry edge since extrema points are expected to occur at $\pm\sqrt{\sigma' + \sigma_b^2}$, where σ' is the reblurring parameter of steerable filters and σ_b is the unknown blur value. Typical example from (ELDER; ZUCKER, 1998)'s work can be seen in Fig. 2.2.

Figure 2.1: Blurry edge and and corresponding 2nd order derivative response. Source: (ELDER; ZUCKER, 1998)

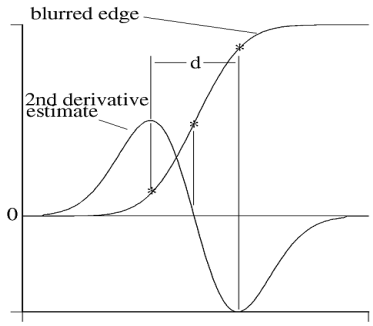
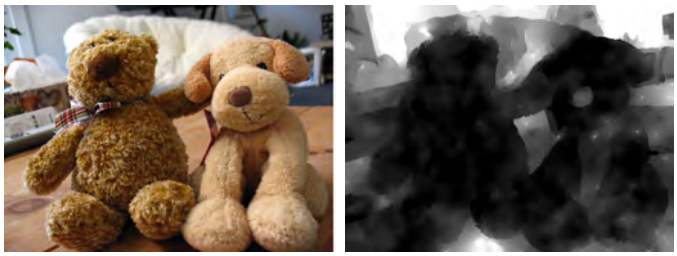


Figure 2.2: From left to right: the synthetic image, detected edge and estimated versus actual blur scale along the edge (ELDER; ZUCKER, 1998).

(BAE; DURAND, 2007) extended (ELDER; ZUCKER, 1998)'s method in order to magnify the existed defocus blur aiming to create an artistic photograph which is called *Defocus Magnification*. (BAE; DURAND, 2007), instead of measuring the distance between extrema, fit the actual pixel response to the multi-scale model

of the second derivative by minimizing the error of fitting using *Least Square Fitting* (LSF). Unlike (ELDER; ZUCKER, 1998)'s work, they propagate the defocus blur amount to whole image since the usage of second order derivatives produces a sparse blur map, in which blur scales are defined only at edge locations. Their propagation is based on non-homogeneous optimization that uses color similarities along the image (similar to (LEVIN; LISCHINSKI; WEISS, 2004)). A typical experimental result for a real image from (ELDER; ZUCKER, 1998) can be seen in Fig. 2.3.

Figure 2.3: The input image and full defocus blur map (BAE; DURAND, 2007)



(TAI; BROWN, 2009) proposed another method for defocus blur map estimation using the *local contrast prior* (LC), which presents a relationship between local image gradients and local image contrasts. They discovered that for blurry regions in images, gradient magnitudes are smaller than the local contrast because of the smoothing effect of the blur; in other words, when the defocus blur increases, the ratio between gradient magnitudes and local contrast increases. Using that information, they suggested a simple yet promising method for defocus blur estimation without using filter-banks or frequency decompositions. However, computing LC for each pixel is not obvious at the homogeneous regions and weak edges due to noise, so that defocus blur will not be defined for each pixel. To tackle this problem (produce a full defocus blur map), they proposed a Markov Random Field propagation scheme that extends the estimated blur amount into the pixels/regions where the local contrast prior cannot be defined.

Later, this work was extended by (SHEN; HWANG; PEI, 2012). In their spatially varying deblurring work, they remove the out-of-focus blur using the estimated defocus blur map. In their method, local contrast priors are used only at edge location, and differently from (TAI; BROWN, 2009), a guided filter (HE; SUN; TANG, 2010) is used for propagation.

(ZHUO; SIM, 2011) proposed a novel method that uses gradient magnitude ratios to estimate the defocus blur amount. The blur estimates are computed at

exact edge locations by taking the ratio of gradient magnitudes of the original image and a re-blurred version of the image. The overview of this approach can be seen in Fig. 2.4. Then, estimated blur scales at edge locations are propagated to whole image using *Alpha Laplacian Matting* (LEVIN; LISCHINSKI; WEISS, 2004), which is a semi-automatic technique for image colorization. Some experimental results can be seen in Fig. 2.5.

Figure 2.4: The overview of the blur estimation approach proposed in (ZHUO; SIM, 2011). Source : (ZHUO; SIM, 2011)

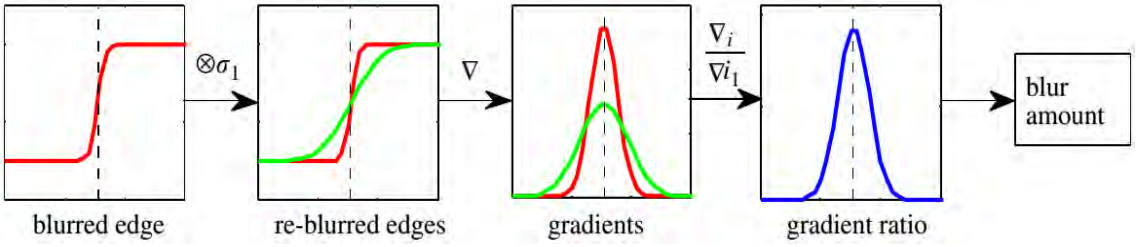


Figure 2.5: Experimental result of (ZHUO; SIM, 2011) for real images. (a)-(c) original images and (b)-(d) are the corresponding defocus blur maps

(a) (b) (c) (d)

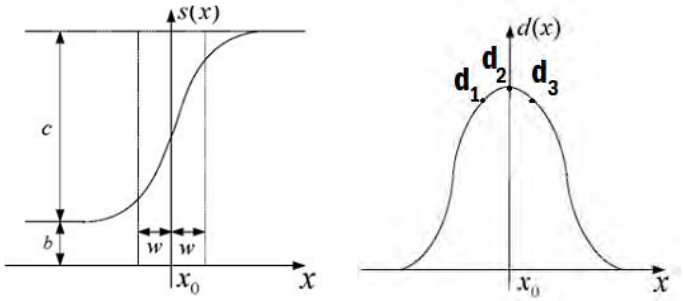
(ZHANG; CHAM, 2012) proposed an edge based defocus blur estimation method for Refocusing and Defocusing. In their method, they modeled edges in a different manner:

$$e(x; b, c, x_0) = cu(x - x_0) + b, \tag{2.2}$$

where b denotes the edge basis and c represents the edge contrast. According to this model, the representation of a typical blurry edge $s(x; b, c, w, x_0)$, which can be obtained by convolving $e(x; b, c, x_0)$ with a 1-D Gaussian kernel $g(x; w) = \frac{1}{\sqrt{2\pi w^2}} \exp\left(-\frac{x^2}{2w^2}\right)$, will be as in Fig. 2.6(a),

$$s(x; b, c, w, x_0) = b + \frac{c}{2}\left(1 + \mathrm{erf}\left(\frac{x - x_0}{w\sqrt{2}}\right)\right), \tag{2.3}$$

Figure 2.6: One dimensional parametric edge model from (ZHANG; CHAM, 2012)



(a) (b)

Then, the defocus blur amount is computed by convolving the blurry edge with a known Gaussian derivative filter $g'_d(x;\sigma_d)$[1], using the assumption that the detected peak lies at $x = 0$, which will result in

$$d(x;c,w;\sigma_d) = \frac{c}{\sqrt{2\pi(w^2+\sigma_d^2)}} \exp\left(-\frac{(x-x_0)^2}{2(w^2+\sigma_d^2)}\right). \tag{2.4}$$

Finally, three point samples d_1, d_2 and d_3 are selected as shown in Fig. 2.6(b) in order to determine all parameters, given by

$$\begin{aligned} c &= d_1\sqrt{2\pi a^2/\ln(l_1)}\, l_2^{\frac{1}{4a}} \\ w &= \sqrt{a^2/\ln(l_1) - \sigma_d^2} \\ x_0 &= 0.5a\ln(l_2)/\ln(l_1) \\ b &= s(x_0) - \frac{c}{2} \end{aligned} \tag{2.5}$$

where $l_1 = \frac{d_1^2}{d_2 d_3}$ and $l_2 = d_2/d_3$. The estimated blur amounts at edge locations w are propagated to non-edge points using a similar approach proposed in (ZHUO; SIM, 2011). Experimental results using real images can be seen in Fig. 2.7.

(TANG; HOU; SONG, 2013) used a similar method as (ZHUO; SIM, 2011) except exploring spectrum contrasts to estimate defocus blur amount at edges. In particular, their method takes into account the chromatic aberration caused by wavelength dependency. More recently, (CHEN; CHEN; CHANG, 2016) proposed a fast defocus blur estimation method based on over-segmentation and transductive inference in order to interpolate estimated blur amount from edge locations to the

[1]The author reports that [1, 3] is a reasonable range for setting σ_d

Figure 2.7: Experimental results for a real image from (ZHANG; CHAM, 2012). (a) the original image and (b) corresponding defocus blur map.

(a) (b)

whole image. (LIU; ZHOU; LIAO, 2016) reformulated the blurry edge considering that the left and right sides of the edge can have different defocus blur amounts (i.e. left σ_l and right σ_r), which is given by,

$$i(x) = bu(-x) * g(x; \sigma_l) + (a + b)u(x) * g(x; \sigma_r), \tag{2.6}$$

with

$$a = \frac{|\sum_{(x,y)\in R_L} i(x, y) - \sum_{(x,y)\in R_R} i(x, y)|}{rl}, \tag{2.7}$$

$$b = \frac{\min\{\sum_{(x,y)\in R_L} i(x, y), \sum_{(x,y)\in R_R} i(x, y)\}}{rl} \tag{2.8}$$

where R_L and R_R denote small $(r \times l)$ size windows on each side of the edge point along the gradient direction.

To compute unknown blur scales σ_l and σ_r, the image is reblurred by multi-scale Gaussian functions, whose standard deviations σ_1, σ_2, ... , σ_N are known. Then, using the gradient magnitude ratio approach similar to (ZHUO; SIM, 2011), $N - 1$ estimates are computed for each edge point (σ_{l_1}, σ_{l_2}, ... , $\sigma_{l_{N-1}}$ and σ_{r_1}, σ_{r_2}, ... , $\sigma_{r_{N-1}}$). Later, these estimates are combined by solving a linear system, which produces a single value for each unknown σ_l and σ_r. Finally, (ZHUO; SIM, 2011)'s approach is adopted to produce full blur map, by using a simplification version of the original image (which is based on structure-texture decomposition (STD)), instead of using the original image, which might produce visible edges on objects in the final blur maps.

2.2 Patch-Based Methods

The second class of defocus blur estimation methods typically explore image patches to estimate the local blur amount. In particular, some of these methods work in the frequency domain, exploring mostly the convolution theorem.

Chakrabarti et al. (CHAKRABARTI; ZICKLER; FREEMAN, 2010) pioneered the exploration of the convolution theorem for blur identification (both defocus and motion). Their work starts with a definition of localized frequency representation assuming that the blur scales are constant in any local neighborhood:

$$f_i[x] = W_i[x] \exp(-2\pi j \langle \omega_i, x \rangle) \tag{2.9}$$

where $W_i[x] \in \{0, 1\}$ is a symmetric window function with limited spatial support, and ω_i is the frequency defined in $\Re^2$. Then, these filters are applied to an image I which results in the corresponding responses $I_i[x] = (I * f_i)[x]$, where the set of values $\{I_i[x]\}_i$ are the Fourier decomposition of a window that centered at location x. Then, using the assumption that h_x (local blur scale) is constant in a local neighborhood, it is derived

$$I_{b_i}[x] = (I * (h_x * f_i))[x] + (\eta * f_i)[x] \tag{2.10}$$

where η is the noise term as described in Eq 1.2.

Since estimation of blur kernels h_x is under-determined from the blurry input image I_b, the authors assume a Gaussian distribution of image gradients as a statistical prior model for the latent sharp image I.

$$I^{\nabla}[x] = (\nabla * I)[x] \sim \mathcal{N}(0, \sigma_g^2). \tag{2.11}$$

Since Eq. 2.10 is linear, I can be written as

$$I_{b_i}^{\nabla}[x] = f_i * I_{b_i}^{\nabla} = (h_x * I^{\nabla}) * f_i + (\nabla * n) * f_i, \tag{2.12}$$

Then, the blur kernels are estimated with a conjunction of MAP approach defined

in (LEVIN et al., 2011).

$$P(\{I_{b_i}^{\nabla}[x]\}_i|h_x) = \prod_i \mathcal{N}(I_{b_i}^{\nabla}[x]|0, \sigma_g^2\sigma_{h_xi}^2 + \sigma_{ni}^2) \tag{2.13}$$

where $\sigma_{h_xi}^2$ is the blur spectrum and σ_{ni}^2 is the noise spectrum. However, even though this formulation showed a good approximation for blur estimation, they claimed that it is not useful for spatially varying case, and instead of choosing a single Gaussian with variance σ_g^2, they proposed to use a continuous range for image gradient variances, tying together the scale parameters within each spatial neighborhood. More precisely, they used

$$P(\{I^{\nabla}[x]\}_{x\in\eta}) = \int P_s(s) \prod_{x\in\eta} \mathcal{N}(I^{\nabla}[x]|0, s)ds \tag{2.14}$$

where η is a small spatial neighborhood, s is the scale parameter, $P_s(s)$ is the probability distribution on s, which is choosen as $\mathcal{S} = (0, s_{max})$. Then the optimal conditional likelihood analogous to the simple Gaussian case (Eq. 2.13) is defined as

$$P(\{I_{b_i}^{\nabla}[x]\}_i|h_x) \propto \arg\min_{s\in\mathcal{S}} \prod_i \mathcal{N}(I_{b_i}^{\nabla}[x]|0, s\sigma_{h_xi}^2 + \sigma_{ni}^2) \tag{2.15}$$

Finally, the optimal value $\widehat{s}$ that maximize Eq. 2.15 is found iteratively using

$$\widehat{s} = \left(\sum_i \rho_i(\widehat{s})\right)^{-1} \sum_i \rho_i(\widehat{s}) \frac{|I_{b_i}^{\nabla}[x]|^2 - \sigma_{ni}^2}{\sigma_{h_xi}^2} \tag{2.16}$$

where

$$\rho_i(\widehat{s}) = \left(1 + \frac{\sigma_{ni}^2}{\widehat{s}\sigma_{h_xi}^2}\right)^{-2} \tag{2.17}$$

Zhu et al. (ZHU et al., 2013) extended Chakrabarti et al. (CHAKRABARTI; ZICKLER; FREEMAN, 2010)'s method claiming that in many cases the global maximum point in Eq. 2.15 does not correspond to the latent h_x. Therefore, they proposed to use a fixed point iteration, calculating optimal h_x and s iteratively with the other variable fixed.

To do that, they generalize the system to handle any kind of blur kernel model, fitting an exponential function of r_b, which is the scale value in the given blur domain $\sigma_{h_xi}^2$ with relatively small interval $\Delta r_b = 0.1$ for $r_b \in [0, 8]$. Then, the optimal $\widehat{r_b}$ is generated by minimizing the following function using the gradient

descent algorithm.

$$\widehat{r_b} = \arg\min_{r_b} \sum_i \left(\frac{|I_{b_i}^{\nabla}[x]|^2}{s\widehat{\sigma}_{h_{xi}}^2(r_b) + \sigma_{ni}^2} + \log(s\widehat{\sigma}_{h_{xi}}^2(r_b) + \sigma_{ni}^2) \right) \tag{2.18}$$

Finally, in order to make a coherent blur map, an energy function is proposed which includes the probability of blur scale estimates and smoothness constraints.

Later, D'Andres et al. (D'ANDRES et al., 2016) proposed a machine learning approach to solve Eq. 2.13. More precisely, the authors proposed to use of Regression Tree Fields (JANCSARY et al., 2012) for learning (and for testing) the blur scales through a simple training data, which was produced manually by reblurring sharp images with known disk kernels. Their work is seminal amongst the defocus blur estimation methods since they produced a realistic ground truth data, which is explained in Sec. 5.1.

In addition, Shi et. al. (SHI; XU; JIA, 2015) proposed a method to estimate defocus blur amount at small regions that span 3-9 pixels (which is called just noticeable blur) based on sparse representation and image decomposition. Their method bases on understanding small image blur via sparse representation of an external data. More precisely, they showed that decomposition of local image patches into dictionary atoms in an additive manner shows quantitatively and as well as visually different results on clear and JNB patches. In a related topic, Vu and colleagues (VU; PHAN; CHANDLER, 2012) presented an approach for estimating the local sharpness, combining spatial and spectral information in a patch-based approach as well. (YI; ERAMIAN, 2016) proposed a Local Binary Pattern (LBP) based sharpness region segmentation scheme using the observation that local blurry image patches have fewer specific LBPs than sharp regions. Using this information, they estimate a sharpness map adopting image matting from (LEVIN; LISCHINSKI; WEISS, 2004) and multi-scale inference.

2.3 Chapter Conclusion

In this chapter, we investigated seminal and recent defocus blur estimation methods. As for edge-based defocus blur estimation methods, mostly start with sparse blur maps, where the corresponding defocus blur amount is defined only at edge points or non-homogeneous regions. Then a full (dense) blur map is obtained

via sparse blur maps by using a propagation algorithm (e.g. Markov Random Field propagation or edge aware filtering). In patch-based defocus blur estimation methods, full blur maps are estimated directly through mostly exploring convolution theorem (when working in the frequency domain) with a Bayesian estimation or a machine learning approach.

In general, it is possible to obtain closed-form models for the blur parameter at image edges, which makes edge-based methods an attractive choice. Among those methods, a popular strategy is to reblur the input image with (one or more) Gaussian kernels, but there is no such mathematical analysis or discussion on how to select such kernels. Also, edge-based methods require interpolation schemes to obtain the full blur map from the sparse one. Several existing approaches explore the Laplacian matting (LEVIN; LISCHINSKI; WEISS, 2004) for that purpose, which is computationally costly and produces artifacts in the blur map at image edges.

In this dissertation, we propose an edge-based method for defocus blur estimation using multiple reblurred versions of the input image. We present an analysis on the impact of the reblurring scale, and propose an adaptive scheme for locally selecting the adequate scale. We also explore a fast guided filter to obtain the full blur map, and extend our method to deal with video sequences, which are barely explored in the literature. Although any defocus blur methods could be applied in a frame-wise manner to a video sequences, this strategy tends to present inconsistencies along time because temporal coherence is not addressed. In this dissertation, we present a simple and fast approach for imposing temporal coherence to defocus blur estimation in video sequences using Kalman Filters.

3 DEFOCUS BLUR ESTIMATION FOR STILL IMAGES

Defocus blur is a natural result of the limitation of focal range of optical lenses. Therefore, for a given camera setting, rays coming from scene points that lie on the focal distance F will converge exactly at the same point on the focal plane. However, rays coming from objects at other distances will not converge to the same point, but to a patch on the sensor plane, referred to as Circle of Confusion (CoC) (PENTLAND, 1987).

A defocus image can be formulated as convolution of the in-focus (sharp) original image and defocus function (characterized by a Point Spread Function - PSF) plus noise (as in Eq. 1.2), and the goal of blur estimation methods is to estimate $\sigma(x, y)$ for each pixel in the image, for a given PSF. In this dissertation, we follow the traditional edge-based pipeline for blur estimation: assuming a Gaussian PSF, we first estimate a sparse map at edge locations, and then propagate the blur information to all pixels.

3.1 Initial Edge-Based Sparse Blur Estimation

3.1.1 Theoretical foundations

Let us consider a 1D signal f containing a sharp edge with amplitude a and offset b at location $x = 0$:

$$f(x) = au(x) + b, \tag{3.1}$$

where u is the unit step function.

Considering a uniform blur, characterized by a spatially - invariant Gaussian PSF $g(x; \sigma_b)$ with variance σ_b^2, the blurred (observed) edge f_b in a continuous-domain signal is given by

$$f_b(x) = f(x) * g(x; \sigma_b), \tag{3.2}$$

so that its derivative is given by

$$\begin{aligned} f_b'(x) &= (f(x) * g(x; \sigma_b))' = f(x) * g'(x; \sigma_b) \\ &= g(x; \sigma_b) = \frac{a}{\sqrt{2\pi\sigma_b^2}} \exp\left(-\frac{x^2}{2\sigma_b^2}\right) \end{aligned} \tag{3.3}$$

If $f_{b_1}(x)$ is a re-blurred version of $f_b(x)$ with another Gaussian kernel $g(x;\sigma_1)$ whose scale parameter σ_1 is known, then the ratio R_g of the derivatives at the edge location is given by

$$R_g = \frac{|f'_b(0)|}{|f'_{b_1}(0)|} = \sqrt{\frac{\sigma_1^2 + \sigma_b^2}{\sigma_b^2}}, \tag{3.4}$$

so that the blur parameter σ_b of the observed image can be obtained as

$$\sigma_b = \frac{1}{\sqrt{R_g^2 - 1}}\sigma_1. \tag{3.5}$$

In fact, this approach was the core of (ZHUO; SIM, 2011)'s defocus blur estimation method.

3.1.2 Mathematical Model of a Blurry Edge

The formulation presented so far allows the local blur estimation using one reblurred version of the observed (blurry) signal $f_b(x)$. A simple extension involves the use of two reblurred versions. More precisely, if $f_{b_1}(x)$ and $f_{b_2}(x)$ are reblurred versions of f_b using Gaussian kernels with variances σ_1^2 and σ_2^2, respectively, the ratio R_g of the derivatives can be computed explicitly as

$$R_g(x) = \frac{|f'_{b_1}(x)|}{|f'_{b_2}(x)|} = \sqrt{\frac{\sigma_b^2 + \sigma_2^2}{\sigma_b^2 + \sigma_1^2}} \exp\left(-\frac{(\sigma_2^2 - \sigma_1^2)x^2}{2(\sigma_b^2 + \sigma_2^2)(\sigma_b^2 + \sigma_1^2)}\right). \tag{3.6}$$

If we evaluate $R_g(x)$ at the exact edge location (i.e. at $x = 0$) and solve Eq. 3.6 for σ_b, we obtain

$$\sigma_b = \sqrt{\frac{\sigma_2^2 - R_g(0)^2\sigma_1^2}{R_g(0)^2 - 1}}, \tag{3.7}$$

assuming that $\sigma_2 > \sigma_1$.

The formulation using one or more reblurred versions of f_b was explored by several methods, such as (ZHANG; CHAM, 2012; KRIENER; BINDER; WILLE, 2013; JIANG et al., 2013; ZHANG et al., 2016; MAHMOUDPOUR; KIM, 2016; CHEN; CHEN; CHANG, 2016). Although the idea is simple and fast, it is important to note that none of these papers provide guidelines on how to select the reblurring parameter(s). It is worth noting that the validity/accuracy of Eq. 3.6 and Eq. 3.7 is affected mainly by three factors when dealing with natural images: i) noise, ii)

accuracy of edge location, and iii) influence of neighboring edges, which might lead to conflicting choices for sigma reblurring parameter. In this work, we first show how σ_1 and σ_2 impact the estimation of the actual blur σ_b, and then present an approach for selecting those parameters adaptively.

The first comment about Eq. 3.7 is that numerical instabilities might arise when $R_g(0) \approx 1$, since it yields small values in the denominator. Hence, selecting $\sigma_2 \gg \sigma_1$ is a good choice, since it yields $R_g(0) \gg 1$.

In the typical image formation model presented in Eq. 1.2, there is also a noise term η added to the blurred image/signal. For isolated edges with zero-mean additive Gaussian noise, the use of larger reblur parameters σ_1 and σ_2 is capable of removing stronger noise, since Gaussian kernels are low-pass filters for which the cut-off frequency decreases as its variance increases.

Also, the formulation that leads to Eq. 3.7 assumes that both $f'_{b_1}(x)$ and $f'_{b_2}(x)$ were evaluated at the correct edge location $x = 0$. In practice, they are evaluated at the locations produced by an edge detector, which might not correspond to the exact edge locations of the image. Hence, Eq. 3.7 holds only approximately, and the error increases with the edge location imprecision. More precisely, the error is smaller when the exponential constant $\frac{(\sigma_2^2-\sigma_1^2)x^2}{2(\sigma_b^2+\sigma_2^2)(\sigma_b^2+\sigma_1^2)}$ is also smaller, which happens when $\sigma_1 \approx \sigma_2$ and/or both σ_1 and σ_2 are large.

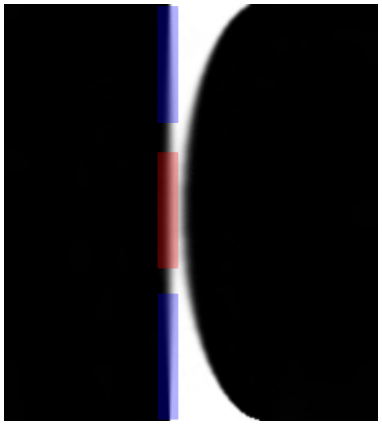
Finally, the core assumption used to obtain Eq. 3.7 is that the sharp underlying signal f contains a single edge. If f contains more than one edge, Gaussian smoothing with a sufficiently large kernel will eventually cause edge merging according to the well-known scale-space theory (WITKIN, 1983). More precisely, the degradation caused by interfering edges depends on the combination of the original blur with the reblur parameters, given by $\sigma_b^2 + \sigma_1^2$, which is the variance of the two successive convolutions with Gaussian kernels ($g(x;\sigma_b)$ and $g(x;\sigma_1)$). The same analysis holds for the second reblur with σ_2. In fact, results with synthetic images shown in (ZHUO; SIM, 2011) report loss of accuracy as the distance from neighboring edges decreases, particularly when the original blur σ_b is larger. In that sense, it is advisable to use a smaller value for σ_1 and σ_2 in order to minimize the influence of neighboring edges.

Based on these mostly conflicting considerations, we can conclude that choosing an adequate value for σ_1 and σ_2 globally is not possible: for isolated edges, larger values better handle edge mis-localization issues and noise; on the other hand,

smaller values should be chosen when nearby edges are present, to avoid edge interference.

In order to illustrate the impact of reblurring scales on edge-based defocus blur estimation methods, let us consider the blurry synthetic image shown in Fig. 3.1, which has vertically varying blur parameter σ_b (in Eq. 3.2) starting from top with $\sigma_b = 0.5$ and increases linearly up to $\sigma_b = 3$ in the middle then drops back linearly again to $\sigma_b = 0.5$. In the top and bottom of the image (highlighted with blue), the edges are well separated, but in the middle (highlighted with red) the structure is thin. We have also produced noisy versions of Fig. 3.1 using zero mean additive Gaussian noise $\eta = 1.275$[1] and $\eta = 2.55$.

Figure 3.1: Artificially blurred synthetic image.



For those three images, as in related papers that explore similar ideas with two or more reblurrings (ZHUO; SIM, 2011; KRIENER; BINDER; WILLE, 2013; MAHMOUDPOUR; KIM, 2016; CHEN; CHEN; CHANG, 2016), we conducted a set of experiments while keeping $\sigma_1 = 1$ fixed and focuses on a the impact of reblurring parameter σ_2. Then, we evaluated the Mean Absolute Error (MAE) of the blur estimate along the top and bottom of the image (blue regions in Fig. 3.1), the central portion (red region in Fig. 3.1) and along the whole left (vertical) edge, using different (global) values for σ_2. The results, shown in Table 3.1, indicate that (as we stated before) choosing an adequate value for σ_1 and σ_2 globally is not possible. It can also be observed that larger values for σ_2 yield better results in the blue regions, which contain little edge interference (and the difference increases as more noise is added), while smaller values for σ_2 are better in the red region

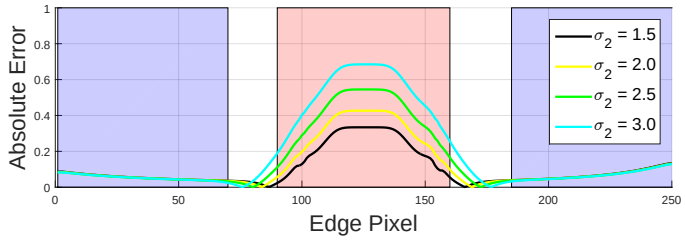
[1]From this point on, this symbol will denote the standard deviation of zero-mean additive Gaussian noise.

(stronger edge interference).

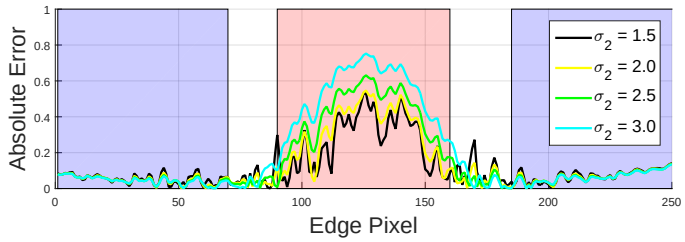
Table 3.1: MAE computed at different edge regions of Fig. 3.1 (Blue/Red/Overall) using $\sigma_1 = 1$ and different values for σ_2 and varying noise contamination.

	Noise Level		
σ_2	$\eta = 0$	$\eta = 1.275$	$\eta = 2.55$
1.5	0.065/**0.224**/**0.103**	0.065/**0.266**/**0.122**	0.081/**0.270**/0.150
2.0	0.064/0.299/0.124	0.061/0.312/0.129	0.067/0.315/**0.145**
2.5	**0.063**/0.399/0.153	0.059/0.404/0.153	0.060/0.412/0.165
3.0	**0.063**/0.521/0.191	**0.057**/0.524/0.189	**0.056**/0.533/0.199

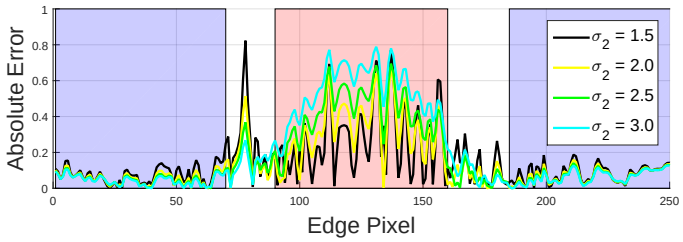
Figure 3.2: (a) Blur estimation errors along the vertical left edge of Fig. 3.1 using $\sigma_1 = 1$ and different values for σ_2. (b)-(c) Analogous result using noisy versions ($\eta = 1.275$ and $\eta = 2.55$) of Fig. 3.1.



(a)



(b)



(c)

In fact, Fig. 3.2 illustrates these behaviors more clearly, by showing the per-pixel blur estimation error along the vertical edge (scanned from top to bottom).

In this dissertation, we proposed two different methods for sparse blur estimation using adaptive scale selection. The first one explores the estimated edge locations using 1D image profiles, whereas the second one uses image gradients directly.

3.1.3 Sparse Blur Map Estimation Using 1D Oriented Profiles

The first proposed method starts with an edge map of the input image I_b computed with the well-known Canny edge detector (CANNY, 1986) (but other edge detection algorithms could be used instead). We also compute the convolution of I_b with 2D isotropic Gaussian kernels having scale parameters $\sigma_1 < \sigma_2 < \cdots \sigma_N$ (where $\sigma_i = 0.75 + 0.25i$ for $i = 1, 2, 3, .., ..., N = 9$), obtaining the multi-scale images $I_{g_i}(x)$ and the corresponding smoothed gradient vectors $\nabla I_{g_i}(x)$. Given the local edge orientation θ (extracted from gradient vectors of the first scale), we extract multi-scale 1D profiles of $\|\nabla I_{g_i}(x)\|$ along θ, centered at each edge point using windows of size $W_h = 11$.

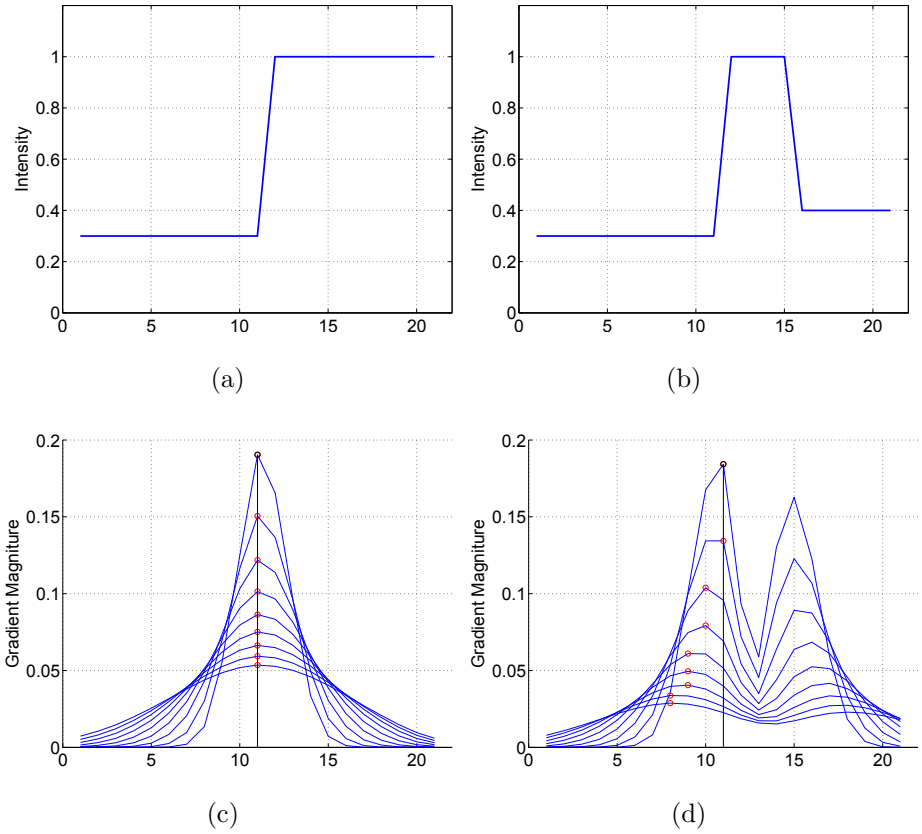
For an isolated edge, the scale-space location (WITKIN, 1983) of the edge should be stable, meaning that the local maxima locations of $\|\nabla I_{g_i}(x)\|$ should not change as σ increases. On the other hand, if there are neighboring edges orthogonal to the local contour orientation, they tend to be merged as σ increases. Figs. 3.3(a) and 3.3(b) illustrate these two scenarios, and the corresponding multi-scale gradient magnitudes along the profile are shown in Figs. 3.3(c) and 3.3(d), respectively. As it can be observed, the local maxima positions of $\|\nabla I_{g_i}(x)\|$ are not affected for the isolated edge scenario, but they start to shift towards the neighboring edge in the second scenario as σ increases, since the interfering edge start to get into the support of the Gaussian kernels.

To find the exact edge location and its support in the scale-space (i.e., the maximum σ value for which there is no interference), we first seek local minima within a neighborhood of the estimated edge location[2] of each different scale individually. Then we find the global maximum between two local minima around the edge (depicted with red points in Fig. 3.3), so that for each scale σ we have an edge location x_σ.

To determine how many scales are reliable enough for defocus blur computa-

[2]the estimated edge location is always at the center of the extracted 1D profile.

Figure 3.3: (a) A preserved edge profile and its gradient profile (c), (b) an edge profile with neighboring edge interference and its gradient profile (d)



tion without causing edge interference, we evaluate how x_σ changes as σ increases. A given scale σ_i is validated if the corresponding edge location x_{σ_i} $(i > 2)$ is closer than a certain distance T from edge location at the smallest blur scale x_{σ_1}, i.e., if $|x_{\sigma_i} - x_{\sigma_1}| < T$ (we have set $T = 2$ based on experiments).

At the end of this process, we have for each estimated edge the maximum number of scales $N_{\max} < N$, as well as the refined edge locations x_{σ_i}, for $i = 1, ..., N_{\max}$. Although we could use any of these scales to estimate the original blur σ_b using Eq. 3.7, we observe that *any* pair of re-blur parameters σ_i and σ_j can be used. In fact, a simple extension of Eq. 3.6 leads to

$$R_{g_{i,j}} = \frac{\|\nabla I_{g_i}(x_{\sigma_i})\|}{\|\nabla I_{g_i}(x_{\sigma_j})\|} = \sqrt{\frac{\sigma_b^2 + \sigma_j^2}{\sigma_b^2 + \sigma_i^2}}, \tag{3.8}$$

so that

$$\sigma_b = \sqrt{\frac{\sigma_j^2 - R_{g_{i,j}}^2 \sigma_i^2}{R_{g_{i,j}}^2 - 1}}. \tag{3.9}$$

Since we have $N_{\max}$ re-blurs for a given edge pixel, we can get $N_{\max}(N_{\max}-1)/2$ estimates for σ_b using Eq. 3.9. However, our experiments indicated that using adjacent scales σ_i and σ_{i+1} produces good results[3], so that we actually obtain $N_{\max}-1$ estimates for each edge pixel. Finally, we combine these estimates in a robust manner by computing the α-trimmed mean, which consists of removing the first and last elements of the sorted elements (which is possibly related to outliers), and computing the mean value of the remaining samples. The estimated blur is given by

$$\sigma_e = \underset{i}{\mu_\alpha}(\sigma_b^i), \tag{3.10}$$

where σ_b^i are the estimates using adjacent scales σ_i and σ_{i+1}.

Figure 3.4: Blur estimation errors along the left edge of the noisy version ($\eta = 1.275$) of the image depicted in Fig. 3.1 using $\sigma_1 = 1$ and different values for σ_2 with comparisons to our 1D adaptive scale selection method.

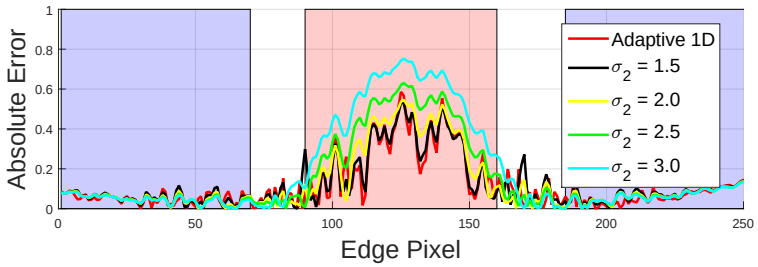


Fig. 3.4 illustrates the effect of the adaptive 1D adaptive scale selection scheme applied to the noisy version ($\eta = 1.275$) of the image depicted in Fig. 3.1, as well as a comparison with other (fixed) values for σ_2. As it can be observed from Table 3.2, the proposed method present the smallest error. In fact, our defocus blur estimation method was published in (KARAALI; JUNG, 2014). Despite the good results obtained with this method, the process of estimating 1D oriented profiles for each edge point (which might require interpolation depending on the orientation) is rather time consuming. Aiming to reduce execution time, we also devised another approach that works directly with image gradients.

[3]Please note that reblurring steps are far enough to each other to deal with aforementioned numerical instabilities, besides that alpha-trimmed mean is capable of to eliminate outliers.

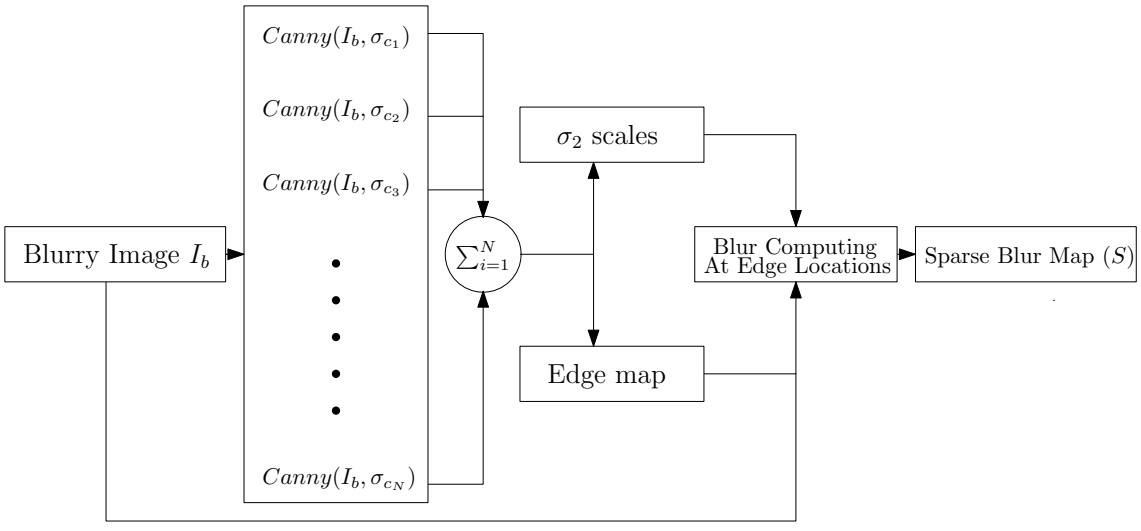
Table 3.2: MAEs computed along the left edge of the noisy version ($\eta = 1.275$) of the image depicted in Fig. 3.1 using $\sigma_1 = 1$ and different values for σ_2 with comparisons to our 1D adaptive scale selection method.

σ_2	Noise Level: $\eta = 1.275$
Adaptive 1D	**0.116**
1.5	0.122
2.0	0.129
2.5	0.153
3.0	0.189

3.1.4 Sparse Blur Map Estimation Using 2D Multi-scale Image Gradients

The core of the second proposed method is to obtain a multi-scale edge map, which is used to adaptively select the local reblurring scale at each edge pixel and obtain a sparse blur map. An overview of the proposed method is presented in Fig. 3.5, and the steps are detailed next.

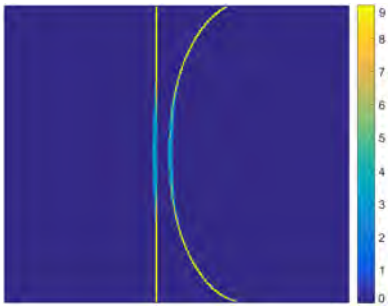
Figure 3.5: Overview of sparse defocus blur map estimation method based on multi-scale image gradients.



The blur model presented so far applies to signal discontinuities, so the first step is to obtain an edge map of the image, as in the previous method. Also, since it involves convolving the signal with Gaussian kernels, the well-known Canny edge detector (CANNY, 1986) seems a natural choice. The main idea is to identify at which scales each edge is detected. For that purpose, our edge point selection

algorithm starts with applying the Canny detector to the observed (blurred) image I_b with known standard deviations $\sigma_{c_1} < \sigma_{c_2} < \cdots < \sigma_{c_N}$ (where N is the number of smoothing scales for Canny edge detector), obtaining the multiscale edge maps $E_{b_i}(x) = Canny(I_b(x), \sigma_{c_i})$.

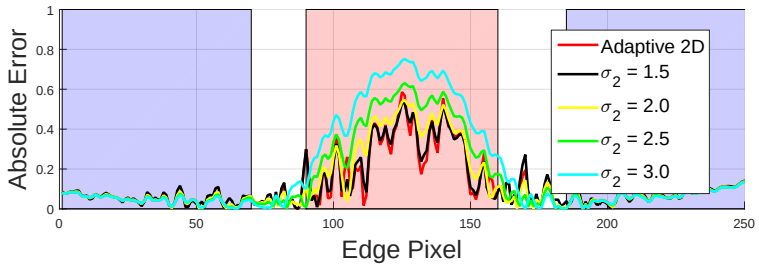
Figure 3.6: Intersection of edge points along the Canny scales on the image depicted in Fig. 3.1.



Isolated edges tend to persist for several scales, whereas nearby edges tend to blend in coarser scales. This phenomenon can be seen in Fig. 3.6, which shows the sum of edge points in all Canny scale (i.e. the number of overlapping scales for each edge pixel). In the thin structure (middle), there are nearby interfering edges, so that the overlap is smaller than the top and bottom of the image. The multi-scale approach is used for two main purposes: selecting edge points and obtaining the local reblurring scale for each edge point. To select the final edge map, we retrieve edge points that were arisen in at least N_r adjacent Canny scales, to ensure some scale-space stability. For each of those points, we also retrieve the largest Canny scale $\sigma_{c_{max}}$ for which it was considered an edge. In fact, this number provides an estimate of edge proximity, since the location of nearby edges (according to maximum of first derivatives or zero-crossings of second derivative) start to change progressively as the Gaussian kernel at one edge affects the other, until they eventually merge.

In our experiments, we used $N = 9$ scales for the multi-scale edge detection scheme, starting with $\sigma_{c_1} = 1$ up to $\sigma_{c_9} = 5$, with steps of 0.5. We also selected $N_r = 4$, so that an edge is validated if it persists for at least $\Delta\sigma = 2$ scales. Although selecting $\sigma_{2_i} = \sigma_{c_{max_i}}$ for each edge pixel i seems an intuitive choice (and it works well for synthetic images), our experiments indicated that using $\sigma_{2_i} = 0.5\sigma_{c_{max_i}}$ yields considerably better results for natural images. Fig. 3.7 shows the mean absolute errors using four different (global) reblurring parameters for σ_2 and our adaptive scheme computed along the whole edge. The MAE obtained using our 2D scheme

Figure 3.7: Blur estimation errors along the left edge of the noisy version ($\eta = 1.275$) of the image depicted in Fig. 3.1 using $\sigma_1 = 1$ and different values for σ_2 with comparisons to our 2D multi-scale image gradients method.



was 0.1163, which is very close to the MAE using the (more costly) version using 1D profiles (0.1162).

3.2 Connected Edge Filter

An initial blur map can be obtained using either 1D adaptive scale selection method, described in Sec. 3.1.3, or the approach that explores multi-scale image gradients directly, presented in Sec 3.1.4. In both of them, the initial blur value is computed independently for each edge pixel. However, edges typically do not appear isolated in natural images: they tend to form contours, which are related to object boundaries. In this work, we assume that connected components in the edge map relate to the same object, and that the blur value tends to vary smoothly along the contour. Let us consider a connected component $\mathcal{C}$ that is formed by N_c pixels p_1, ..., p_{N_c}, and α_j denotes the desired blur estimate at each pixel ($i = 1, ..., N_c$). Given the initial blur estimates σ_{b_i} provided by Eq. 3.7, our goal is to find the set of regularized blur values $\alpha_1, ..., \alpha_{N_c}$ that minimize an error function E that accounts for both the local individual error at each pixel and spatial continuity of neighboring edges in $\mathcal{C}$. The proposed energy E is given by

$$E = \sum_{i=1}^{N_c} \left[\xi_i (\alpha_i - \sigma_{b_i})^2 + \sum_{j=1}^{N_c} \psi_{ij} (\alpha_i - \alpha_j)^2 \right] \tag{3.11}$$

where ξ_i is a confidence value for each estimated defocus blur σ_{b_i}, so that when ξ_i is large the refined estimate α_i tends to be closer to the initial estimate σ_{b_i}.

In general, isolated edges allow the selection of larger reblurring scales (for both initial blur estimation methods), which better handles edge mis-localization

and noise, and tends to produce more reliable estimates. Based on this fact, for initial blur estimation method using 1D oriented profiles we used the selected number of reblurring scales as ξ_i, and for the approach based on 2D Image Gradients, we used $\xi_i = c_{max_i}$ as the confidence parameter.

In Eq. 3.11, the weights ψ_{ij} control the pairwise connections of pixels i and j along the contour, so that choosing larger values for ψ_{ij} yields more similarity between α_i and α_j. The proposed connected edge filter (CEF) takes into account edge proximity for enforcing smoothness, and we use

$$\psi_{ij} = \begin{cases} \Psi, & \text{if } p_i \text{ and } p_j \text{ are neighbors} \\ 0, & \text{otherwise} \end{cases}, \tag{3.12}$$

considering an 8-connected neighborhood, where Ψ is a constant. Larger values for Ψ yield more smoothing along the edge, whereas smaller values lead to a stronger weight for the data fidelity term.

The minimization of E is straightforward, leading to a sparse, symmetric diagonally-dominant (SDD) linear system (there are at most nine non-zero entries in each line of the coefficients matrix), which can be solved efficiently. In fact, recent methods (SPIELMAN; TENG, 2014) can solve such systems in nearly $\mathcal{O}(n \log^{1/2} n)$ time. Since the edge map is already very sparse, and n is the number of edge pixels in a given connected component, this step has low computational complexity (furthermore, each connected component can be solved in parallel).

Figure 3.8: Blur estimation errors using 1D adaptive scale selection and 2D multi-scale image gradients methods along the left edge of the noisy version ($\eta = 1.275$) of the image depicted in Fig. 3.1 and their corresponding Connected Edge Filtered (CEF) versions.

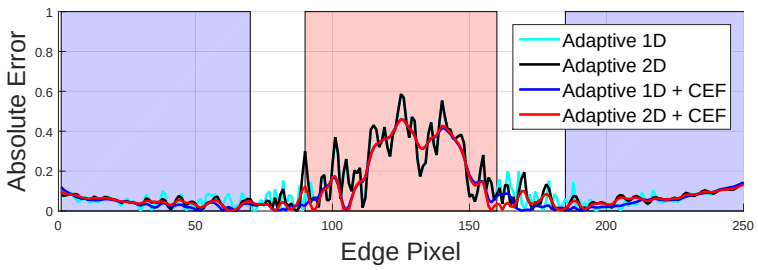


Fig. 3.8 illustrates the effect of coupling the CEF with the proposed adaptive scale selection methods for the image illustrated in Fig. 3.1 with noise level $\eta = 1.275$. It can be observed that Connected Edge Filtering smooths the estimated blur scales

along the contour. If Ψ is too small, the filtered estimates would be very similar to the initial ones. On the other hand, if Ψ is very large, the filtered blur estimates would be smoother but with less fidelity to the original estimates. Choosing the optimal value for Ψ is a difficult task, and in this dissertation we used $\Psi = 10$ in all experiments. The MAE computed along the whole edge for the 1D adaptive scale selection and 2D multi-scale gradients methods, and their corresponding filtered versions can be seen in Table. 3.3.

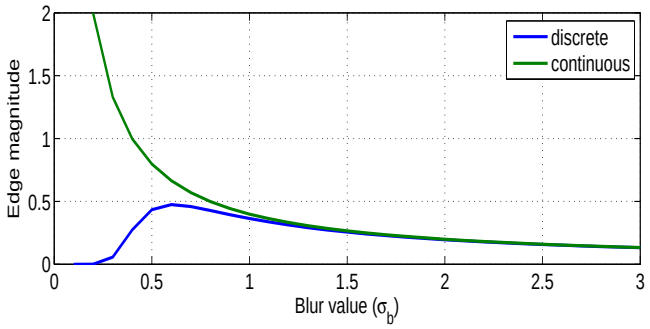
Table 3.3: MAEs computed using 1D adaptive scale selection and 2D multi-scale image gradients methods along the left edge of the noisy version ($\eta = 1.275$) of the image depicted in Fig. 3.1 and their corresponding Connected Edge Filtered (CEF) versions.

σ_2	Noise Level: $\eta = 1.275$
Adaptive 1D	0.1162
Adaptive 2D	0.1163
Adaptive 1D + CEF	**0.0990**
Adaptive 2D + CEF	0.1044

3.3 Sparse Blur Map Refinement

Another important issue in the original formulation presented in (ZHUO; SIM, 2011) is that it is valid only in the continuous domain. For digital images, the domain is discrete and the convolution result shown in Eq. 3.3 is valid only asymptotically when $\sigma_b \rightarrow \infty$, as shown in Fig. 3.9. As a consequence, the blur estimation using Eq. 3.7 contains discretization errors, particularly when σ_b is small.

Figure 3.9: Comparison of the convolution of a (1D) step edge with the derivative of a Gaussian in the continuous and discrete domains.



In order to deal with the discretization issue, we introduce a correction factor that accounts for the error in the discretization process. Given the estimate σ_e obtained with Eq. 3.10, we can write

$$\sigma_e = \sigma_b + g\left(\frac{1}{\sigma_b}\right), \tag{3.13}$$

where $g(x)$ is the correction factor such that $g(x) \approx 0$ if $x \approx 0$, i.e. the error decreases as σ_b increases. Using a first order approximation of g, we can write

$$\sigma_e \approx \sigma_b + \frac{\beta}{\sigma_b}, \tag{3.14}$$

where β is a constant. To find β, we used Least Square Fitting (LSF) based on a synthetic image with a single edge with spatially varying blur (from 0 to 5). Since discretization errors are very large for $\sigma_b < 0.5$ (see Figure 3.9), we only used blur values in the range [0.5 5], obtaining $\beta = 0.0832$.

Finally, when Eq. 3.14 is solved with respect to σ_b, we obtain the final blur estimate

$$\sigma_b' = \frac{\sigma_e + \sqrt{\sigma_e^2 - 4\beta}}{2}. \tag{3.15}$$

Figure 3.10: Blur estimation errors using 1D adaptive scale selection and 2D multi-scale image gradients methods along the left edge of the noisy version ($\eta = 1.275$) of the image depicted in Fig. 3.1 and their corresponding Connected Edge Filtered (CEF) and Refined (Ref.) versions.

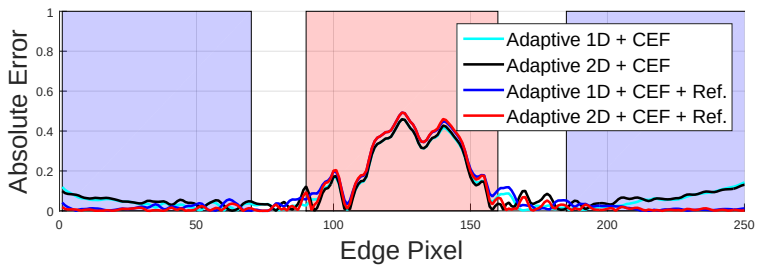


Fig. 3.10 shows an analogous result to Fig. 3.8, but also adding the refinement step to the pipeline. It can be observed that the inclusion of the refinement step lower the errors along low blur regions of the image shown in Fig. 3.1, which correspond to the leftmost and rightmost portions of the plot. The MAE computed along the whole contour for the 1D adaptive scale selection and 2D multi-scale image gradients methods, as well as their corresponding filtered and refined versions, can be seen in

Table 3.4: MAEs computed using 1D adaptive scale selection and 2D multi-scale image gradients methods along the left edge of the noisy version ($\eta = 1.275$) of the image depicted in Fig. 3.1 and their corresponding Connected Edge Filtered (CEF) and Refined (Ref.) versions.

σ_2	Noise Level: $\eta = 1.275$
Adaptive 1D	0.1162
Adaptive 2D	0.1163
Adaptive 1D + CEF	0.0990
Adaptive 2D + CEF	0.1044
Adaptive 1D + CEF + Ref.	0.0897
Adaptive 2D + CEF + Ref.	**0.0816**

Table. 3.4.

3.4 Defocus Blur Map Propagation

The algorithms so far compute the defocus blur amount only at edge locations. However, the formation process affects the entire image, so that the estimated defocus blur amounts should be propagated to the whole image.

In (ZHUO; SIM, 2011; PI et al., 2012; ZHANG; CHAM, 2012; TANG; HOU; SONG, 2013; LIU; ZHOU; LIAO, 2016; ZHANG et al., 2016) the Laplacian-based image colorization algorithm (LEVIN; LISCHINSKI; WEISS, 2004) was selected. However, it is slow and the edges are visible in the blur map, even in regions with uniform blur (see some results in Chapter 5). In this dissertation, we explore a very fast guided filter (GASTAL; OLIVEIRA, 2011) to propagate the sparse blur map.

The presented sparse map estimation algorithms are based on a binary edge image M, which is obtained with Canny or our adaptive Canny detector, and they produce a sparse blur map S at the edge locations, such that $S(x, y)$ is only defined when $M(x, y) = 1$. If we consider the propagation scheme as a diffusion process given by the heat equation

$$\frac{\partial B}{\partial t} = \gamma \Delta B, \tag{3.16}$$

with initial conditions given by the sparse map

$$B(x,y,0) = \begin{cases} S(x,y), & \text{if } M(x,y) = 1 \\ 0, & \text{otherwise} \end{cases}, \tag{3.17}$$

the solution of Eq. 3.16 is the convolution with a Gaussian kernel for which the variance increases with t. Since the Gaussian kernel is not an interpolating one, the normalization B/M' is required, where M' is the diffusion of M with the heat equation.

A clear drawback of the heat equation is that diffusion is isotropic, and the stationary solution is a constant function. As noted in (PARIS; KORNPROBST; TUMBLIN, 2009) and explored in (LANG et al., 2012a), when anisotropic diffusion schemes (that prevent diffusion across edges) are used, the stationary solution in the discrete case is asymptotically equivalent to an edge-aware guided filter.

Edge-aware guided filtering is basically a technique that smooths images while preserving edges, and the idea of edge preserving while image smoothing can be traced back to (AURICH; WEULE, 1995)'s work on Non-Linear Gaussian Filters, which was named as *Bilateral Filtering* later by (TOMASI; MANDUCHI, 1998). Edge-aware filtering can be also used for sparse data up-sampling when feature correspondences are defined sparsely in an image. Among the several existing edge-aware methods, the domain transform approach presented in (GASTAL; OLIVEIRA, 2011) seems a good choice for our interpolation scheme. Its core is to perform domain transforms in 1D signals based on color and spatial similarity (using spatial and range parameters σ_{sp} and σ_{ra}, as in traditional bilateral filters) such that convolutions with a fixed filter in the transformed domain act as an edge-preserving filter in the original signal. For images, the filter is applied iteratively (and in alternation) in the rows and columns. When box filters are used, the approach leads to a very fast implementation, regardless of the values for σ_{sp} or σ_{ra}.

Let $\mathcal{D}_f(J, I)$ denote the joint domain-filtered version of image J using image I as reference (i.e., the content in I is used to propagate the information in image J). We first filter the input blurry image by using the edge-aware filter itself, i.e. we compute $I'_b = \mathcal{D}_f(I_b, I_b)$, to remove fine textures that might compromise information propagation. In fact, this simplification procedure has the same goal as the structure-texture decomposition (STD) explored in (LIU; ZHOU; LIAO, 2016), which is used in conjunction with the (costly) Laplacian-based image colorization

algorithm (LEVIN; LISCHINSKI; WEISS, 2004) to generate the full blur map.

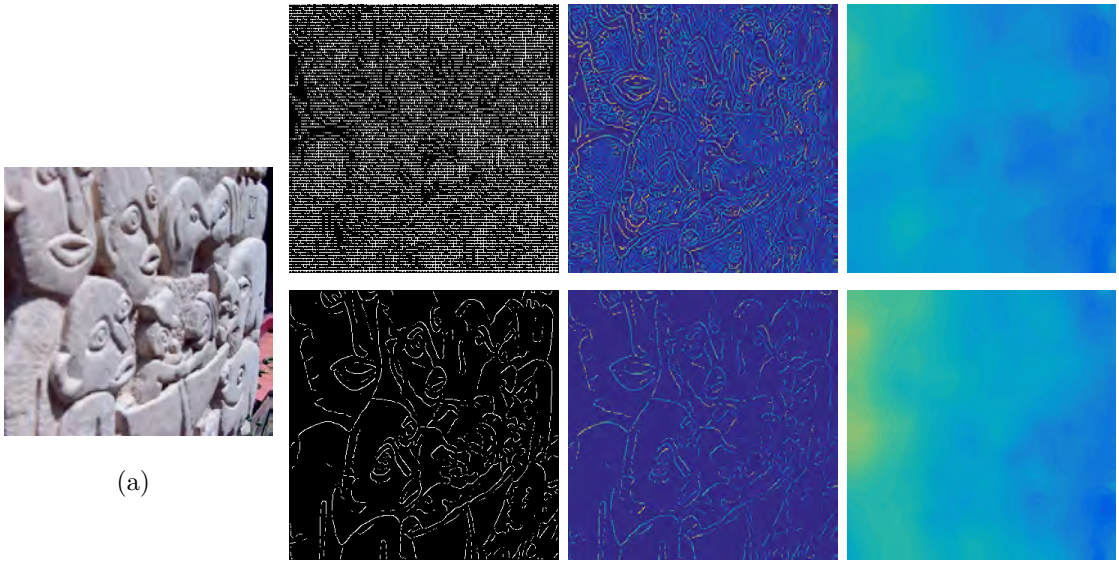
The simplified image I'_b is then used as a guide to generate the full blur map through

$$B = \frac{\mathcal{D}_f(S, I'_b)}{\mathcal{D}_f(M, I'_b)}, \tag{3.18}$$

recalling that M is the binary edge map and S is the sparse blur map, which contains information only at non-zero entries of M.

Regarding parameter selection, we used $\sigma_{sp} = 7$ and $\sigma_{ra} = 0.5$ to obtain the simplified image I'_b (assuming pixels values in the range $[0,\ 1]$ at each color channel). For data propagation, we used $\sigma_{sp} = \min\{R, C\}/8$, $\sigma_{ra} = 3.75$, where R and C are the dimensions of I_b (height and width). Both spatial and range kernel sizes are larger when performing data interpolation than when simplifying the reference image, since the goal of simplification is to perform local smoothing (to remove fine texture/noise), whereas for data interpolation the sparse blur map must be propagated to pixels far from the edges.

Figure 3.11: (a) A naturally blurry image from (D'ANDRES et al., 2016)'s dataset. (b) Step by step full blur map estimation using our 1D adaptive scale selection and 2D multi-scale image gradients methods. From left to right: Edge map, sparse defocus blur map and full blur map of 1D adaptive scale selection method (for first row) and 2D multi-scale image gradients method (for second row).



(a)

(b)

To illustrate the proposed method, we chose an image from (D'ANDRES et al., 2016)'s dataset Fig. 3.11(a) and the intermediate results image using the proposed approaches are shown in Fig. 3.11(b). The top row of shows the results of

the 1D adaptive scale selection method, while the bottom row shows the 2D multi-scale image gradients method. The edge maps are shown on the left, the sparse blur maps on the middle, and the full (interpolated) blur map on the right. For more images and comparisons please refer to Chapter 5.

4 DEFOCUS BLUR ESTIMATION ON VIDEO SEQUENCES

With the proposed defocus blur estimation methods, we are able to generate promising and visually coherent defocus blur maps for still images. However, if one of our methods or any other ones that we revised in Chapter 2 are applied to a video in a frame-by-frame manner, the resulting defocus map video may contain temporal visual inconsistencies, since the defocus blur is estimated without any information from previous or subsequent frames. Therefore, the inclusion of temporal coherence is very important to obtain a smoother defocus blur map (in time), without jittering.

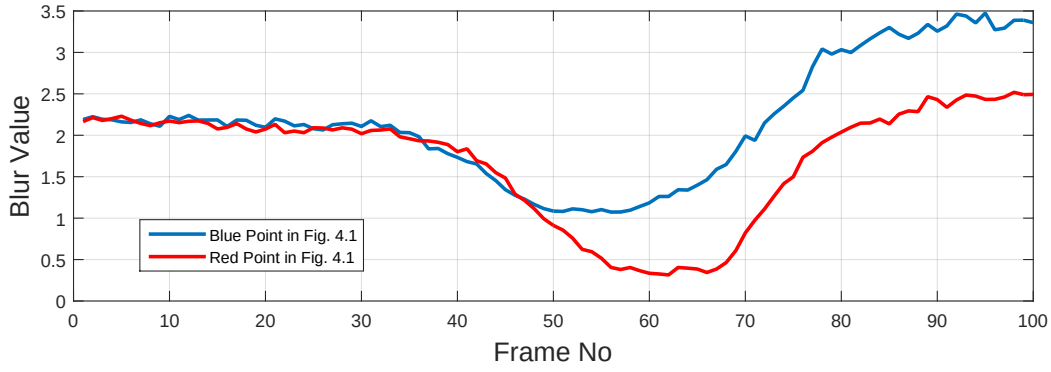
A typical example of blur flickering along time can be seen in Figs. 4.1 and 4.2. Fig. 4.1 shows a few frames extracted from a video sequence in which the focal plane changed in time along with two reference points, while Fig. 4.2 shows the blur values in time for these two reference points computed by the proposed 2D gradient scheme applied to each frame of the video sequence. It is important to point out that although our algorithm produces visually coherent results on individual frames, yet when we choose a pixel and investigate the blur amount along time, we observe temporally incoherent blur variations (jitter), as illustrated in Fig. 4.2.

Figure 4.1: Video sequence acquired with a shallow DOF camera

Temporal coherence in visual processing has a long history and has been applied to various applications such as optical flow (VOLZ et al., 2011; CHIN; KARL; WILLSKY, 1994), disparity estimation using stereo cameras (YANG, 2014) and scene flow (LV et al., 2016), among others. However, to the best of our knowledge it has not been explored in the context of video defocus blur estimation.

Although temporal coherence arises in a variety of computer vision problems, the goal is the same: when the same pixel/region is present in a set of frames, the

Figure 4.2: Estimated defocus blur scales at two reference points (depicted in Fig. 4.1) along time.



desired information (flow, disparity, blur, etc.) tends to vary smoothly; on the other hand, sharp temporal transitions should be allowed during occlusions/disocclusions. In general, identifying the same pixel or region across a set of frames relates to the problems of dense particle tracking or optical flow. In fact, Lang and colleagues (LANG et al., 2012b) presented a generic framework for imposing temporal consistency in a variety of applications using optical flow. However, blurry regions typically present weak textural information, which may compromise the quality of tracking-based methods. To illustrate this behavior, we applied two well known optical flow methods (BROX; MALIK, 2011; REVAUD et al., 2015) to some blurry video sequences. More precisely, in one test we used a blurry video that presents a moving object (a car) and also camera movement. We manually labeled some feature points on the object, which is initially out-of-focus and then we tracked those points by doing particle advection in the optical flow produced by (BROX; MALIK, 2011). The first frame of the sequence, along with the initialized particles, are shown in Fig. 4.3(a). The partial trajectories obtained by particle advection for other frames of the video sequence are illustrated in Fig. 4.3(b) and Fig. 4.3(c), indicating that points are not correctly tracked in time. In particular, the particle that was initialized close to the right headlight of the car (bottom right in Fig. 4.3) diverges in time. A similar test is also conducted in the reverse temporal order to observe the accuracy of tracking from in-focus to out-of-focus transition. In this test, the initial particles are shown in Fig. 4.5(c), and the partial trajectories in Fig. 4.5(b) and Fig. 4.5(a). Even particles that were initialized at good regions, such as the corner of the headlights, diverge in time as focus level of the object changes.

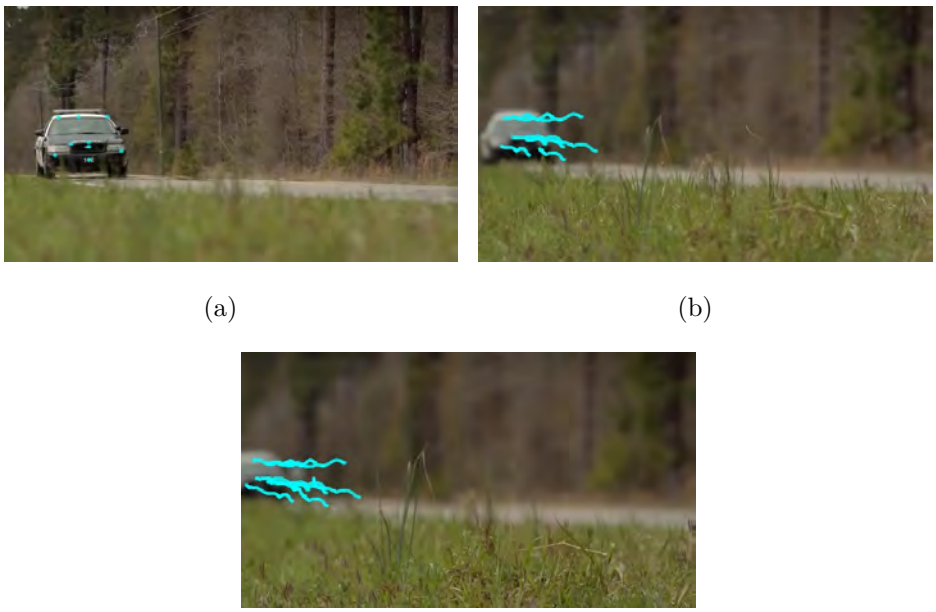
In another experiment, we evaluated visually the results produced by (REVAUD et al., 2015) when applied to a simple short sequence with no noticeable

Figure 4.3: Blurry video frames and trajectory of the tracking points. (a) frame no #1, (b) frame no #34 and (c) frame no #54. Source : a TV Series called UNDER THE DOME©.

(a) (b)

(c)

Figure 4.4: Blurry video frames and trajectory of the tracking points in reverse order. (a) frame no #54, (b) frame no #34 and (c) frame no #1. Source : a TV Series called UNDER THE DOME©.

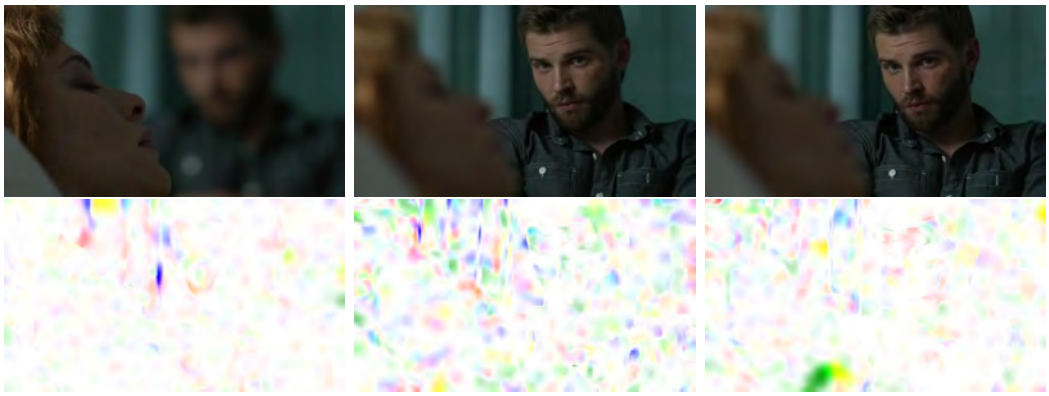


(a) (b)

(c)

camera or object motion, only change in the focal plane. The optical flow should produce low magnitude vectors, but because of the weak textural information due to the blur, results were not as expected. The first row of Fig. 4.5 shows three chosen frames from the blurry video and second row shows the corresponding optical flow vector using Correspondence Visualization method[1]. As it can be observed from the second row of Fig. 4.5, static scene shows some movement (as color for direction and intensity level of the this color for magnitude), possibly caused by defocus blur.

Figure 4.5: Blurry video frames. From left to right : frame #2, frame #51 and frame #80, and corresponding optic flow vectors. Source : a TV Series called UNDER THE DOME©.



Based on the failure results obtained with well known optical flow methods, we looked for other alternatives to impose temporal consistency for defocus blur estimation in video sequences. Despite the existence of some attempts for blur invariant feature extraction (SAAD; HIRAKAWA, 2016), there is no optical flow method that takes blur information into account (according to our knowledge).

As investigated in Chapter 2, there are numerous defocus blur map estimation approaches for still images, but as far as our literature review indicates, there are no methods focused on video sequences. Next we present an approach to generate temporally coherent defocus blur maps from video sequences using the Kalman filter (HOEFFKEN; OBERHOFF; KOLESNIK, 2011), which has been used for a long time in several vision and graphics problems, such as depth estimation from video sequences (MATTHIES; KANADE; SZELISKI, 1989).

[1] <https://hci.iwr.uni-heidelberg.de/Correspondence_Visualization>

4.1 Kalman Filter for Temporal Coherence

The main idea of the proposed temporal consistency approach is to initially apply our blur estimation method to each frame independently, which are considered observations of the actual blur. In a subsequent step, the blur values at each pixel are filtered temporally such that when there are image cues suggesting that the pixel belongs to the same object along several frames, the blur value should vary smoothly. If the pixel under analysis belongs to more than one object within a temporal window, then sharp transitions in the filtered blur value are allowed.

Based on these considerations, we explored Kalman filters to implement temporal coherence, since they offer a closed-form solution to the Bayesian filtering problem, allowing an adaptive selection of the compromise between data (observation) fidelity and smoothing. When applied to scalar values, the Kalman filter involves a state transition model

$$x(t+1) = F_t x(t) + w(t), \tag{4.1}$$

$x(t)$ is the (unobserved) state at frame t, F_t is a constant that models the expected state transition between times t and $t+1$, and $w(t)$ is the process noise. The mathematical formulation of the Kalman filter also involves observation model given by

$$y(t) = H_t x(t) + v(t), \tag{4.2}$$

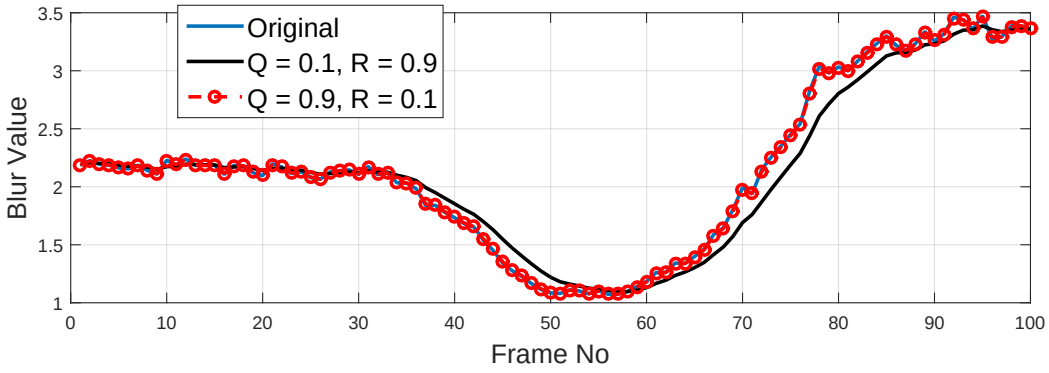
where $y(t)$ is the observation at time t, H_t relates the observation and the state, and $v(t)$ corresponds to measurement noise. Both process noise $w(t)$ and the measurement noise $v(t)$ are assumed to be independent and normally distributed: $w \sim N(0, Q_t)$ and $v \sim N(0, R_t)$, where Q_t and R_t are the corresponding variances.

In the context of temporally coherent blur estimation, $x(t)$ is the desired blur value of a given pixel along time, and $y(t)$ is the output produced by the defocus blur estimation method applied in a frame-wise manner. According to our assumption, which is smooth blur scale transition along the frames for a DoF video sequence, the state transition scalars are constant and equal to one, i.e. $F_t = 1, \forall t$. Also, the best guess for the unknown blur $x(t)$ given only the frame-wise estimate is $x(t) = y(t)$, which means that $H_t = 1, \forall t$ in Eq. 4.2.

The main challenge in the formulation is how to define the state and observa-

tion noise variances R_t and Q_t, which define the balance between the state transition and the observation models. When Q_t is small and R_t is large, the state transition model imposes that $x(t+1) \approx x(t)$, whereas $x(t)$ might be far away from $y(t)$ due to high noise in the observation model. As a consequence, smoothing prevails over data fidelity. On the other hand, the opposite is expected when Q_t is large and R_t is small: the observation model imposes that $x(t) \approx y(t)$, while $x(t+1)$ could be far from $x(t)$ due to high noise in the state transition model. In that case, data fidelity prevails over smoothing. Clearly, those are only the extreme cases, and the compromise between smoothing and data fidelity is highly related to the choices of R_t and Q_t. This compromise can be seen in Fig. 4.6: small process noise variance Q_t enforces higher temporal coherence if the reliability R_t is low (high measurement noise). However, when the measurement noise variance R_t is high Kalman filter results are less smooth.

Figure 4.6: The effect of the state and observation noise variances R and Q at the blue reference point depicted in Fig 4.1



Based on these observations, noise variances, R_t and Q_t should be selected based on the following guidelines:

- The measurement noise variance R_t should be small if the blur value at the pixel of interest is obtained "reliably".

- The process noise variance Q_t, which enforces the temporal coherence, should be small when the pixel appearance does not change in time.

The proposed method for choosing R_t and Q_t adaptively based on the local video content is presented next.

4.1.1 The Measurement Noise Variance

As explored in Chapter 3, the reliability of the blur estimation results depends highly on the edges, since such examples with an isolated edge (e.g. blue regions of Fig. 3.1) yielded highly accurate results. In the Connected Edge Filter defined in Section 3.2, spatial smoothing over the connected edge components was performed by minimizing the energy function given in Eq. 3.11. That term includes a compromise between data fidelity and smoothing, which is similar to the temporal smoothing compromise provided by the Kalman Filter. In that step of our method, the weight ξ_i of data fidelity in a given pixel i was chosen based on the agreement of the multiscale blur estimates c_{max_i}, so that larger values for ξ_i yield more data fidelity.

For temporal smoothing, a similar choice was done. However, in the Kalman Filter formulation data fidelity increases as R_t gets closer to zero, so that R_t should be a non-negative monotonically decreasing function of ξ (the pixel index i was omitted). Although there are several possibilities for this function, we experimentally defined

$$R_t = e^{-\frac{\xi_t}{\chi}} \tag{4.3}$$

where χ is a constant that controls the decay of the exponential (and it is set $\chi = 2$ experimentally), and ξ_t is the data fidelity term in Eq. 3.11 for the pixel under consideration at time t.

Clearly, ξ_t is only computed at edge locations, since it is defined as the number of scales for which the Canny detector produces an edge at the pixel under consideration. To obtain a confidence value at all image pixels, we follow a similar approach to the defocus blur interpolation presented in Section 3.4. More precisely, we applied Eq. 3.18 to the sparse confidence map using the simplified image I'_b as a guide.

4.1.2 The Process Noise Variance

For the process noise $w \sim N(0, Q^{(t)})$, which controls the smoothness, we use a patch similarity measure between frames t and $t+1$ for a given pixel location. More precisely, we initially determine the Mean Square Error (MSE) of a pixel around a

small neighborhood of pixel (x, y) across consecutive frames to estimate Q_t:

$$Q_t = \frac{1}{W_Q} \sum_{(m,n) \in W_Q(x,y)} \|\boldsymbol{I}_{\boldsymbol{b}}^t(m,n) - \boldsymbol{I}_{\boldsymbol{b}}^{t+1}(m,n)\|^2, \tag{4.4}$$

where $W_Q(x, y)$ is a $N_Q \times N_Q$ neighborhood (which is set to 3×3 initially), and $\boldsymbol{I}_{\boldsymbol{b}}^t(m, n)$ is the the 3D vector with the RGB color values (in the range $[0,\ 1]$) of the input blurry image at frame t and pixel (m, n). The formulation measures the similarity of image points around a neighborhood between two consecutive frames. Using a neighborhood in lieu of just a pixel point gives more reliable confidence value to enforce the temporal coherence since similar image points (small MSE) will be evaluated as there will not be a sharp blur transition between frames at that point.

5 EXPERIMENTAL RESULTS AND APPLICATIONS

In this chapter we present experimental results of the proposed deblurring method applied to still images and also to video sequences. We also show applications that can benefit from the estimated blur maps, such as non-blind image deblurring and blur-aware image retargeting. For all methods used in the evaluation, we used the (MATLAB) code and/or data provided by the authors except (ZHANG; CHAM, 2012), and for the sake of fairness at running time comparisons, we have also used a MATLAB version of our code. All experiments are conducted on a PC with an Inter Core i7, 3.50GHz CPU and 16GB RAM machine and it is run on a single thread without using parallelism.

5.1 Evaluation of Blur Estimation Methods

Quantitative evaluation of defocus blur estimation methods is a challenging task. Most methods report their results on (non-standardized) synthetic images. On one hand, the use of synthetic images allows the generation of fully controlled experiments, in which all parameters (e.g. noise, ground truth blur amounts, PSF type, edge location, etc.) are known and can be changed to reproduce specific situations. On the other hand, they are quite simple and do not reflect the complexity of natural images (for instance, regarding noise, edge mis-localization and edge interference, as it was mentioned in Chapter 3). For example (LIU; ZHOU; LIAO, 2016) proposed a validation strategy that consists of artificially blurring a natural image (with known ground-truth PSF), computing the blur estimates for both images (natural and artificially blurred), and then comparing the difference with the ground truth. Although the idea is interesting, one drawback of this approach is that consistent errors in both estimations might cancel out when doing the subtraction.

In this dissertation, we initially evaluate the quality produced by the proposed sparse blur estimation method. Since it is computed at image edges, we use only synthetic images in this initial analysis, and compare our results with other edge-based methods. Next, we evaluate the results of the full blur map estimation, which accuracy is a combination of the sparse estimation with the data propagation scheme. For that purpose, we use a set of natural sharp images that are synthetically blurred (so that the ground truth blur map is known), and we also show comparative results

using a database of naturally blurred images with "known" blur parameters, recently presented by (D'ANDRES et al., 2016). In their database, images were acquired using a Lythro lightfield camera, and the ground-truth blur was in fact estimated from the multiple focal planes by matching reblurred patches of the sharp (all-in-focus) image with the blurry version, and refining the blur map using a regularization method.

5.2 Defocus Blur Estimation Results

5.2.1 Sparse Defocus Blur Estimation

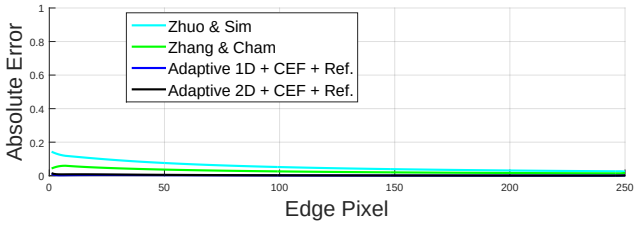
The first test involves a synthetic image containing two vertical stripes with linearly increasing blur amount from top ($\sigma_b = 0.5$) to bottom ($\sigma_b = 3$) and two noisy versions (e.g. $\eta = 1.275$ and $\eta = 2.55$), as depicted in Fig. 5.1[1]. The absolute errors of blur estimates along the vertical contour (rastered from top to bottom) produced by our sparse methods are shown in Fig. 5.2, along with the results produced by (ZHUO; SIM, 2011) and (ZHANG; CHAM, 2012). Those methods were chosen for the comparison because they serve as basis for several other edge-based methods. As it can be observed, the results produced by both of our approaches present lower errors than the two other edge-based methods. More precisely, the MAE of the four evaluated techniques is shown in Table 5.1, corroborating the visual results seen in Fig. 5.2.

Figure 5.1: Artificially blurred edge stripe images with varying noise contaminations: from left to right no noise, $\eta = 1.255$ and $\eta = 2.55$

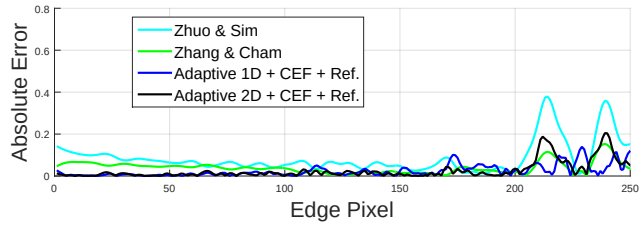
For the second experiment, we used same images shown in Fig. 5.1 but simulating the edge mis-localization problem. For that purpose, we artificially shifted

[1]Please note that, the image with noise level $\eta = 1.275$ cannot be differentiated visually

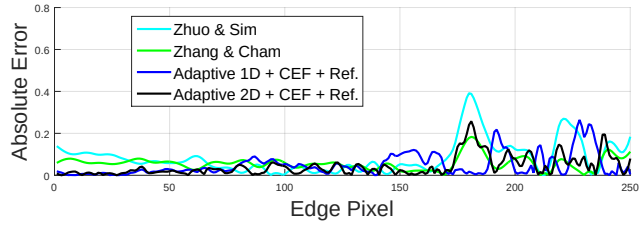
Figure 5.2: Blur estimation errors along the edge of the image depicted Fig. 5.1 for different noise level; (a) $\eta = 0$ (b) $\eta = 1.275$ (c) $\eta = 2.55$. CEF and Ref. stand for Connected Component Filtering and refinement respectively.



(a)



(b)



(c)

Table 5.1: Mean Absolute Errors along the edge of the image depicted Fig. 5.1 for different noise levels. CEF and Ref. stand for Connected Component Filtering and refinement respectively.

	Noise Level		
Method	$\eta = 0$	$\eta = 1.275$	$\eta = 2.55$
Zhuo & Sim	0.0533	0.0860	0.0875
Zhang & Cham	0.0265	0.0395	0.0580
Adaptive 1D+CEF+Ref.	**0.0020**	**0.0248**	0.0492
Adaptive 2D+CEF+Ref.	0.0031	0.0261	**0.0442**

the edge map horizontally to simulate the effect of computing magnitude ratios not exactly at edge locations. As summarized in Table 5.2, for different noise levels and edge mis-localization shifts, the proposed 1D adaptive scale selection method does

Table 5.2: Mean Absolute Errors along the edge of the image depicted Fig. 5.1 for different noise and edge mis-localization levels. CEF and Ref. stand for Connected Component Filtering and refinement respectively.

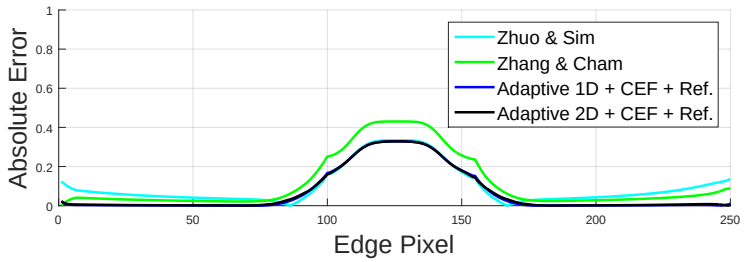
	Noise Level			
Edge Shift	Method	$\eta = 0$	$\eta = 1.275$	$\eta = 2.55$
1px	Zhuo & Sim	1.0461	1.0413	0.9839
	Zhang & Cham	0.0263	0.0406	0.0690
	Adaptive 1D+CEF+Ref.	**0.0020**	**0.0246**	**0.0483**
	Adaptive 2D+CEF+Ref.	0.9321	0.9296	0.9301
2px	Zhuo & Sim	1.7515	1.6153	1.5030
	Zhang & Cham	0.0259	0.0617	0.0888
	Adaptive 1D+CEF+Ref.	**0.0023**	**0.0245**	**0.0480**
	Adaptive 2D+CEF+Ref.	2.4509	2.4049	2.5883

not suffer from the edge mis-localization problem similar to in (ZHANG; CHAM, 2012). This behavior is expected since the 1D adaptive scale selection method explores local maxima to determine the exact edge location, which makes it robust for small edge mis-localization conditions, similarly to (ZHANG; CHAM, 2012). On the other hand, the method proposed by (ZHUO; SIM, 2011), as well as our 2D multi-scale image gradients approach, suffer more due to edge mis-localization issue. This behavior is expected, since the mathematical formulation of both methods assume gradient magnitude ratios computed at edge locations. However, it is important to point out that those results were obtained by artificially introducing mis-localization errors, and the multi-scale edge detection used in our 2D version keeps only scale-persistent edges, which tend to be more accurate.

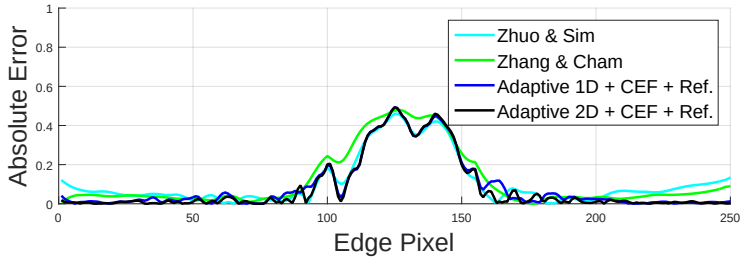
For the final experiment, we used the image shown in Fig. 3.1, which presents also the problem of interfering edges, very common in natural images. The MAE along the left contour is shown in Fig. 5.3. Although other competitive methods look smoother, both of the proposed methods present lower MAEs, since good estimates are preserved by the data fidelity term of our CEF. Clearly, increasing the weight of the regularization term in the CEF will produce smoother results, but the selected parameters were chosen based on a good compromise between smoothness and error. In fact, this can be seen better in Table 5.3: both of the proposed methods are superior to other competitive methods, as well as the methods where a fixed

reblurring parameter is used (see Table 3.1) at all noise levels.

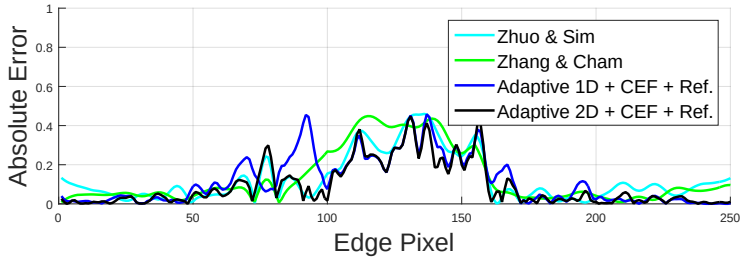
Figure 5.3: Blur estimation errors along the vertical left edge of Fig. 3.1 for different noise levels; (a) $\eta = 0$ (b) $\eta = 1.275$ (c) $\eta = 2.55$. CEF and Ref. stand for Connected Component Filtering and refinement respectively.



(a)



(b)



(c)

These results indicate that both methods for obtaining the initial sparse map present lower errors than other methods. However, the method using 1D adaptive scale selection method[2] is significantly slower than the 2D multi-scale image gradients method. Hence, the remaining results for full blur map evaluation (as well as the applications) will be based on the 2D multi-scale image gradients with Connected Edge Filtering and Refinement will be referred to as "Our Method" from this point on.

[2]Running time of the 1D adaptive scale selection method is highly dependent on the number of edge points, yet our experiments showed that it is 100 times slower than 2D multi-scale image gradients method for 480×640 images

Table 5.3: Mean Absolute Errors along the left edge of the image depicted Fig. 3.1 for different noise levels. CEF and Ref. stand for Connected Component Filtering and refinement respectively.

	Noise Level		
Method	$\eta = 0$	$\eta = 1.275$	$\eta = 2.55$
Zhuo & Sim	0.1021	0.1096	0.1271
Zhang & Cham	0.1150	0.1165	0.1305
Adaptive 1D+CEF+Ref.	**0.0677**	0.0897	0.1147
Adaptive 2D+CEF+Ref.	0.0688	**0.0816**	**0.0912**

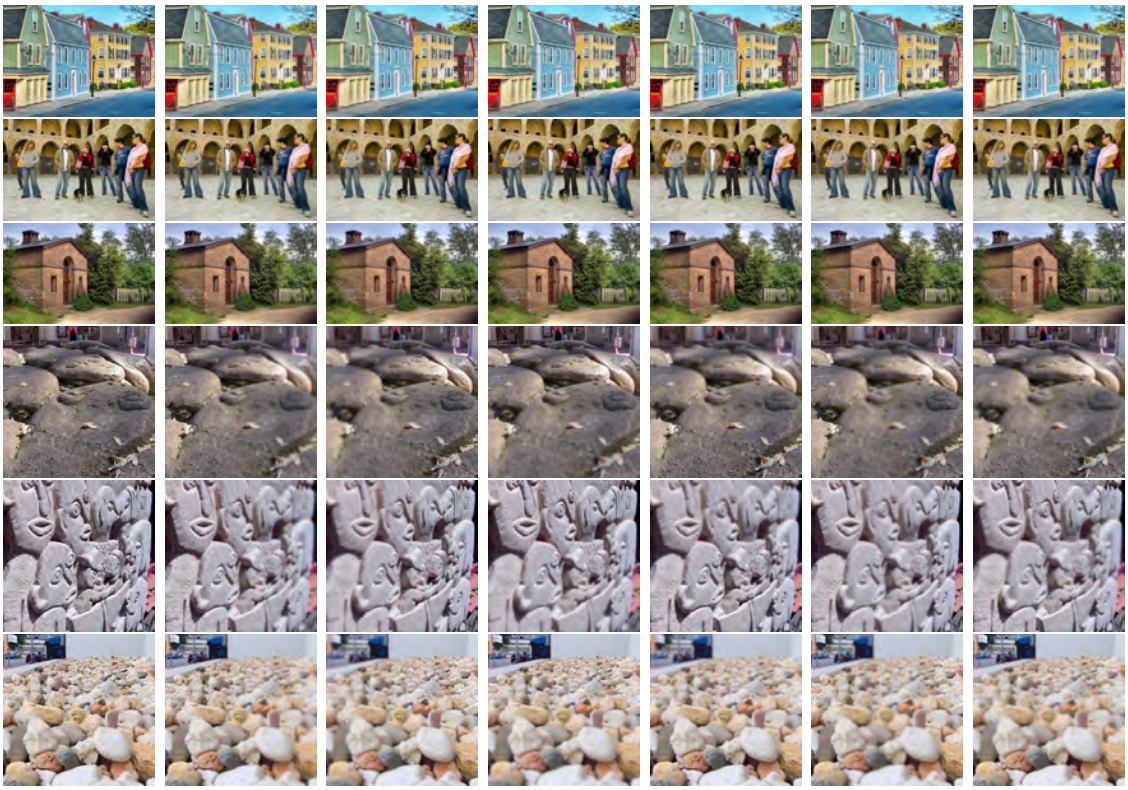
5.3 Full Defocus Blur Estimation

5.3.1 Natural Images with Artificial Blur Maps

Clearly, natural images present more challenges than the simple synthetic images used to test most defocus blur estimation methods, particularly due to the edge content. In this part of the dissertation, we adopted a hybrid validation procedure: we used a set of visually sharp (all objects in focus) images from (RUBINSTEIN et al., 2010) and from (D'ANDRES et al., 2016), and produced blurred versions using six different spatially varying Gaussian PSFs, so that ground truth is known and a quantitative evaluation is possible. More precisely, we used the `Marblehead Mass`, `Umdan` and `Brick House` images from (RUBINSTEIN et al., 2010), and *Image 01*, *Image 04* and *Image 19* from (D'ANDRES et al., 2016). These images are shown in the first column of Fig. 5.4, and the corresponding artificially blurred versions are shown in the following columns.

The first column of Fig. 5.5 shows the six blur maps, called PSF1-PSF6, used to create the blurry versions of Fig. 5.4. The remaining rows, from left to right, correspond to the blur maps for the `Marblehead Mass` image obtained with (ZHUO; SIM, 2011), (ZHANG; CHAM, 2012), (BAE; DURAND, 2007), (TANG; HOU; SONG, 2013), (SHEN; HWANG; PEI, 2012), (CHEN; CHEN; CHANG, 2016) and our method, respectively. Figs. 5.5-5.10 show analogous results for images `Umdan`, `Brick House`, *Image 01*, *Image 04* and *Image 19*, respectively. As it can be observed, the results produced by our approach are visually more coherent with the actual blur maps, not having sharp transition on edges or inconsistent blur scales.

Figure 5.4: Left: full in-focus images. Remaining rows (left to right): artificially blurred versions using known Gaussian PSFs, shown in Fig. 5.5. Source : (RUBINSTEIN et al., 2010) and (D'ANDRES et al., 2016).



An objective evaluation was then performed by computing MAE of the blur maps estimated using our method and the competitive approaches evaluated in the analysis, as summarized in Tables 5.4 to 5.9. It can be observed that the proposed method produced better results than competitive approaches for all blur maps for each of the analyzed images, corroborating the visual results shown in Figs. 5.5- to 5.10.

Figure 5.5: Left column: blur maps used to generate the blurry images on the first row of Fig. 5.4. Remaining columns (left to right): estimated blur maps produced by (ZHUO; SIM, 2011), (ZHANG; CHAM, 2012), (BAE; DURAND, 2007), (TANG; HOU; SONG, 2013), (SHEN; HWANG; PEI, 2012), (CHEN; CHEN; CHANG, 2016), and the proposed method, respectively.

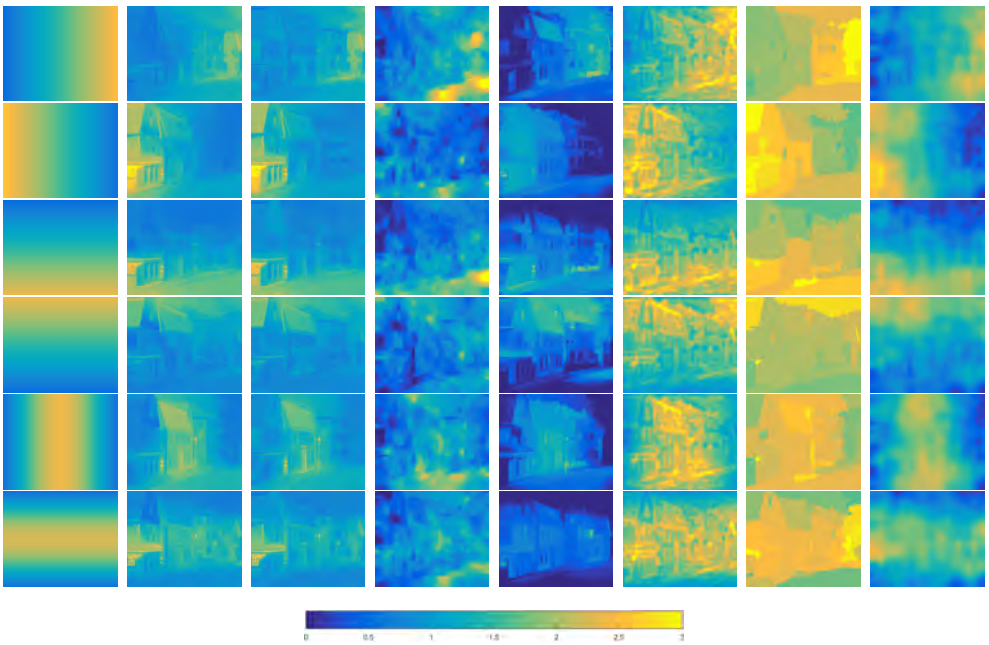


Figure 5.6: Left column: blur maps used to generate the blurry images on the second row of Fig. 5.4. Remaining columns (left to right): estimated blur maps produced by (ZHUO; SIM, 2011), (ZHANG; CHAM, 2012), (BAE; DURAND, 2007), (TANG; HOU; SONG, 2013), (SHEN; HWANG; PEI, 2012), (CHEN; CHEN; CHANG, 2016), and the proposed method, respectively.

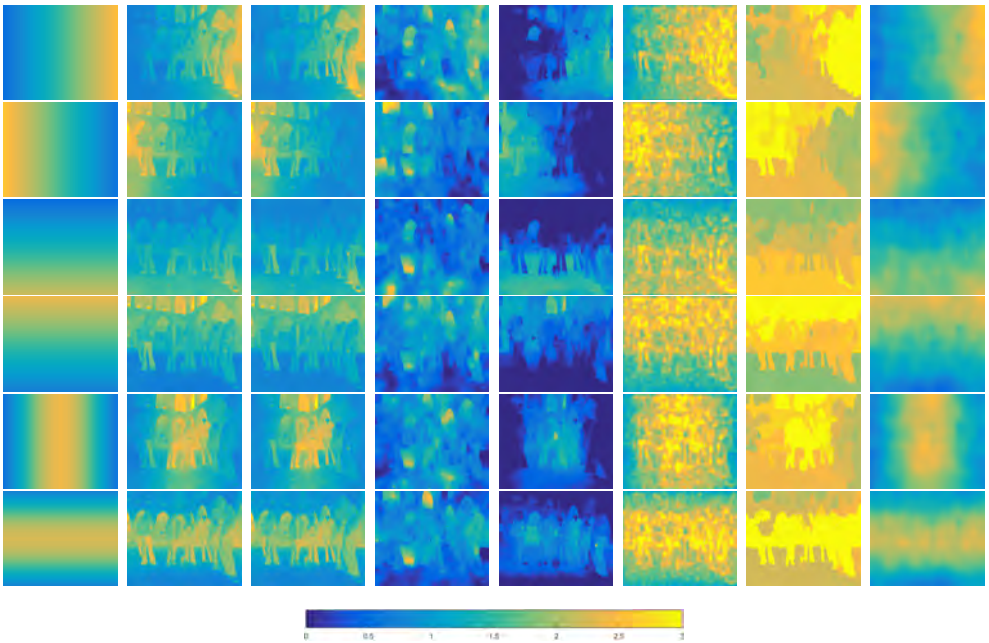


Figure 5.7: Left column: blur maps used to generate the blurry images on the third row of Fig. 5.4. Remaining columns (left to right): estimated blur maps produced by (ZHUO; SIM, 2011), (ZHANG; CHAM, 2012), (BAE; DURAND, 2007), (TANG; HOU; SONG, 2013), (SHEN; HWANG; PEI, 2012), (CHEN; CHEN; CHANG, 2016), and the proposed method, respectively.

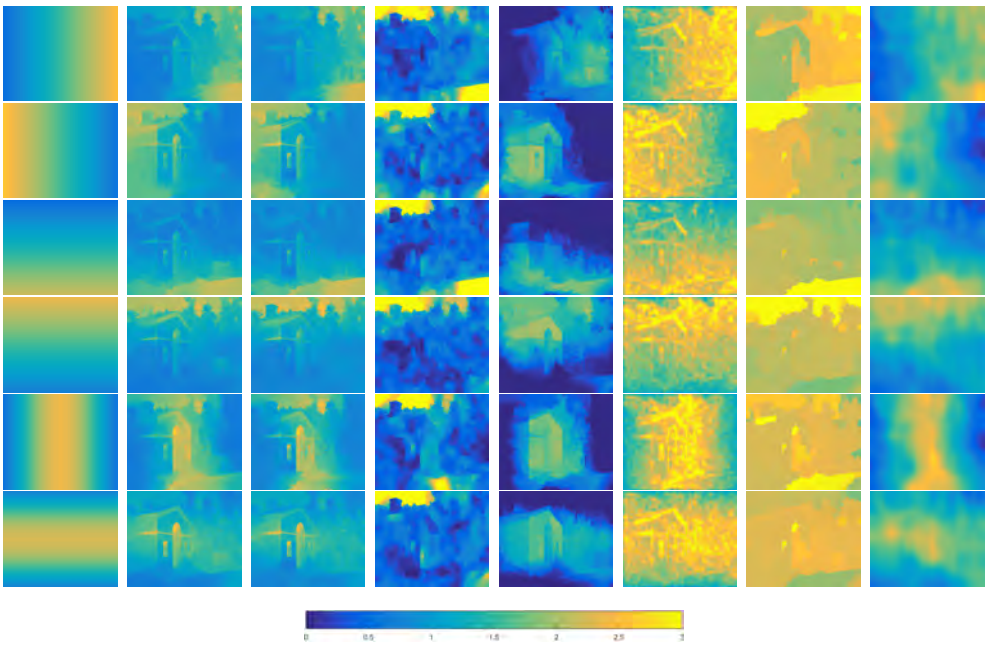


Figure 5.8: Left column: blur maps used to generate the blurry images on the forth row of Fig. 5.4. Remaining columns (left to right): estimated blur maps produced by (ZHUO; SIM, 2011), (ZHANG; CHAM, 2012), (BAE; DURAND, 2007), (TANG; HOU; SONG, 2013), (SHEN; HWANG; PEI, 2012), (CHEN; CHEN; CHANG, 2016), and the proposed method, respectively.

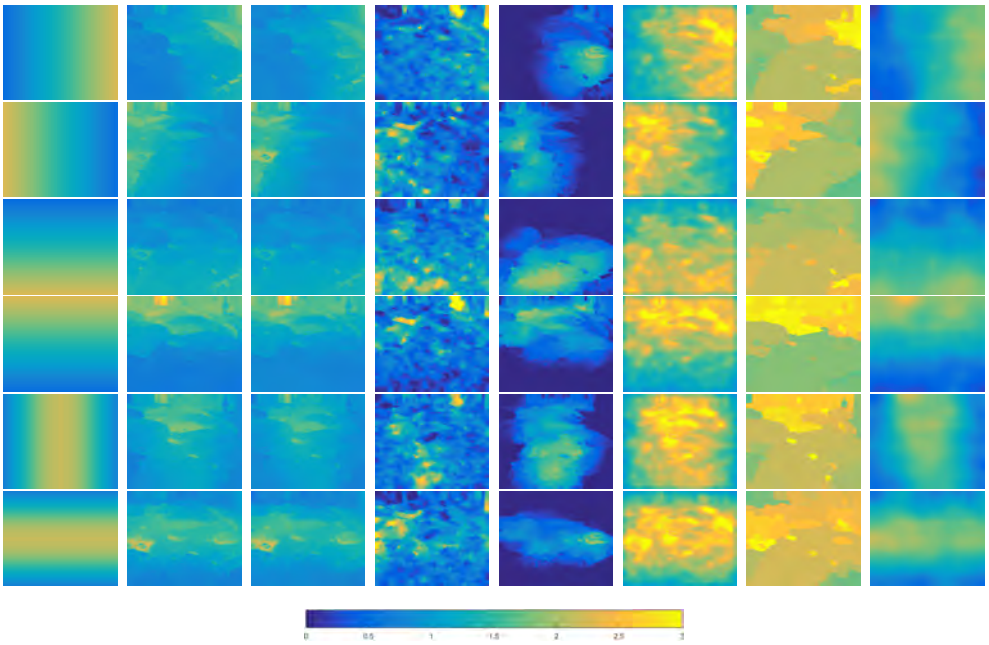


Figure 5.9: Left column: blur maps used to generate the blurry images on the fifth row of Fig. 5.4. Remaining columns (left to right): estimated blur maps produced by (ZHUO; SIM, 2011), (ZHANG; CHAM, 2012), (BAE; DURAND, 2007), (TANG; HOU; SONG, 2013), (SHEN; HWANG; PEI, 2012), (CHEN; CHEN; CHANG, 2016), and the proposed method, respectively.

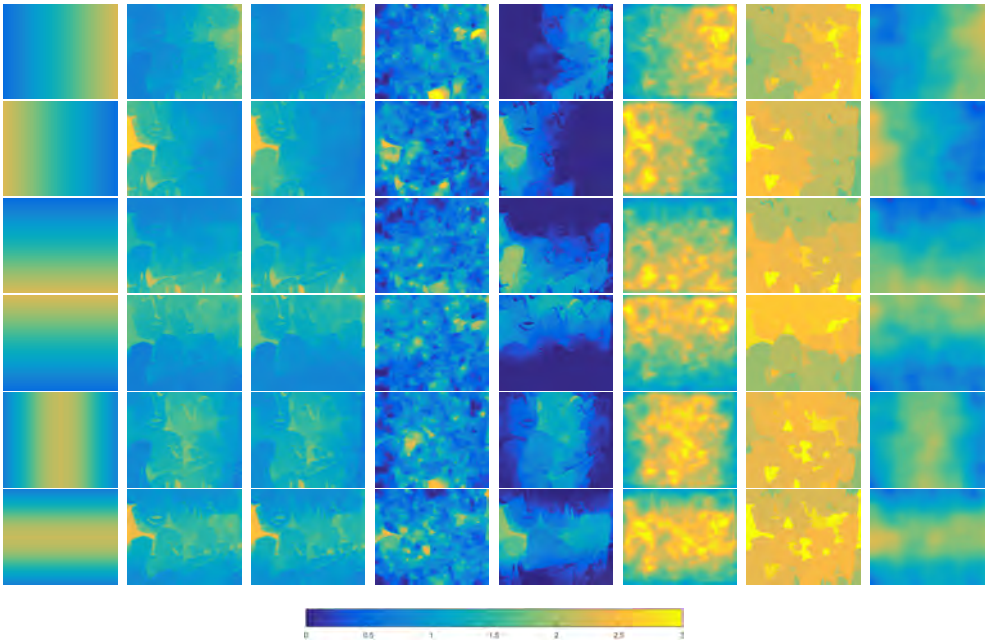


Figure 5.10: Left column: blur maps used to generate the blurry images on the sixth row of Fig. 5.4. Remaining columns (left to right): estimated blur maps produced by (ZHUO; SIM, 2011), (ZHANG; CHAM, 2012), (BAE; DURAND, 2007), (TANG; HOU; SONG, 2013), (SHEN; HWANG; PEI, 2012), (CHEN; CHEN; CHANG, 2016), and the proposed method, respectively.

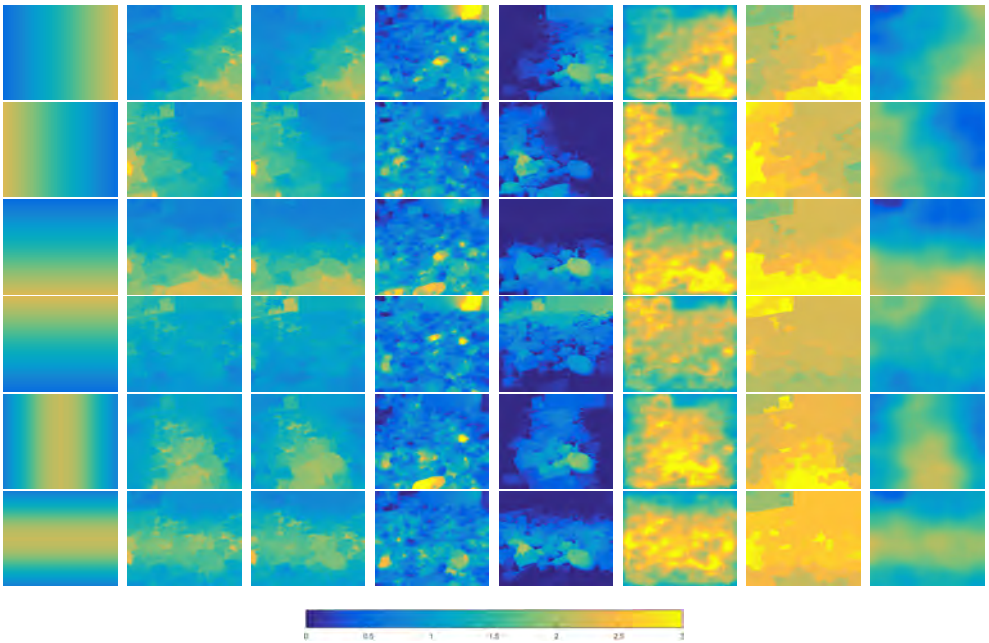


Table 5.4: MAE comparisons on artificially blurred images - PSF1

Image Name	Zhuo&Sim	Zhang&Cham	Bae&Durand	Tang et.al.	Shen et.al.	Chen et.al.	*OurM.*
Marblehead Mass	0.4543	0.5085	0.4952	0.4065	0.3261	0.7208	**0.2803**
Umdan	0.2246	0.2895	0.5837	0.4441	0.5445	1.0339	**0.0667**
Brick House	0.3477	0.3797	0.9315	0.4085	0.6204	0.7905	**0.2040**
Image 01	0.3966	0.4198	0.6424	0.5772	0.5469	0.8024	**0.1854**
Image 04	0.2956	0.3403	0.6221	0.4653	0.5667	0.9068	**0.1380**
Image 19	0.2817	0.3096	0.5837	0.4317	0.6367	0.9432	**0.1923**

Table 5.5: MAE comparisons on artificially blurred images - PSF2

Image Name	Zhuo&Sim	Zhang&Cham	Bae&Durand	Tang et.al.	Shen et.al.	Chen et.al.	*OurM.*
Marblehead Mass	0.3636	0.4376	0.9066	0.5007	0.3442	0.7718	**0.2586**
Umdan	0.2338	0.3006	0.7566	0.4386	0.5438	0.9806	**0.1025**
Brick House	0.3526	0.3928	0.9682	0.5521	0.5677	0.7499	**0.2208**
Image 01	0.3457	0.3716	0.6776	0.4421	0.5406	0.7944	**0.1751**
Image 04	0.2833	0.3231	0.7351	0.5895	0.5441	0.9342	**0.1228**
Image 19	0.2689	0.2916	0.6493	0.6430	0.6298	1.0092	**0.1801**

Table 5.6: MAE comparisons on artificially blurred images - PSF3

Image Name	Zhuo&Sim	Zhang&Cham	Bae&Durand	Tang et.al.	Shen et.al.	Chen et.al.	*OurM.*
Marblehead Mass	0.3552	0.3983	0.6212	0.3747	0.3216	0.7403	**0.2840**
Umdan	0.2373	0.3112	0.6708	0.3831	0.5737	0.9290	**0.1158**
Brick House	0.3263	0.3553	1.0292	0.4515	0.6499	0.8132	**0.2197**
Image 01	0.3799	0.4395	0.5031	0.4830	0.5352	0.7152	**0.2064**
Image 04	0.3145	0.3507	0.6852	0.5104	0.5502	0.9049	**0.1436**
Image 19	0.1749	0.1867	0.5264	0.6272	0.6287	1.1311	**0.1358**

Table 5.7: MAE comparisons on artificially blurred images - PSF4

Image Name	Zhuo&Sim	Zhang&Cham	Bae&Durand	Tang et.al.	Shen et.al.	Chen et.al.	*OurM.*
Marblehead Mass	0.4392	0.4433	0.7414	0.3221	0.3685	0.6838	**0.2660**
Umdan	0.2378	0.2642	0.6969	0.5641	0.5627	0.9954	**0.0734**
Brick House	0.4116	0.4046	0.9242	0.4083	0.6272	0.7587	**0.1941**
Image 01	0.2982	0.3377	0.7223	0.4271	0.5615	0.8292	**0.2007**
Image 04	0.2874	0.3483	0.6163	0.5072	0.5651	0.9262	**0.1411**
Image 19	0.3824	0.3917	0.6748	0.3030	0.7104	0.8776	**0.2532**

Table 5.8: MAE comparisons on artificially blurred images - PSF5

Image Name	Zhuo&Sim	Zhang&Cham	Bae&Durand	Tang et.al.	Shen et.al.	Chen et.al.	*OurM.*
Marblehead Mass	0.4519	0.5209	0.6665	0.4717	0.3402	0.6774	**0.2941**
Umdan	0.2563	0.3020	0.7610	0.6903	0.5324	0.9825	**0.1106**
Brick House	0.3528	0.4075	0.9906	0.4835	0.5759	0.8049	**0.2102**
Image 01	0.4322	0.4925	0.7834	0.4441	0.5096	0.6976	**0.2630**
Image 04	0.3589	0.4171	0.7566	0.3958	0.5275	0.8550	**0.1829**
Image 19	0.3336	0.3486	0.8019	0.7038	0.6298	0.9221	**0.2057**

Table 5.9: MAE comparisons on artificially blurred images - PSF6

Image Name	Zhuo&Sim	Zhang&Cham	Bae&Durand	Tang et.al.	Shen et.al.	Chen et.al.	*OurM.*
Marblehead Mass	0.4247	0.4980	0.7761	0.6967	0.3476	0.7152	**0.2870**
Umdan	0.2783	0.3187	0.9198	0.7038	0.5210	0.9598	**0.0922**
Brick House	0.4309	0.4905	1.2210	0.4418	0.5975	0.6494	**0.2667**
Image 01	0.3979	0.4336	0.7473	0.8250	0.5305	0.7542	**0.2030**
Image 04	0.3562	0.4131	0.6948	0.4591	0.5377	0.8990	**0.1571**
Image 19	0.3050	0.3403	0.7550	0.6729	0.6010	0.9522	**0.2014**

5.3.2 Naturally Blurred Images

This section shows the full blur map estimation results of our method and also competitive techniques for the database presented in (D'ANDRES et al., 2016). Results were evaluated quantitatively through the Mean Absolute Error (MAE) and qualitatively through visual inspection of the produced blur maps. However, it is important to recall that the provided ground-truth blur values are represented as disc PSF parameters (the radius of the disk, to be more precise). Considering that, for methods that assume a Gaussian PSF (such as ours and several of the edge-based methods that follow a similar formulation), a conversion function (kindly provided by the authors of (D'ANDRES et al., 2016)) was used for comparison, and its effect is going to be discussed in Subsection 5.3.3.

The MAE obtained with the edge-based methods (ZHUO; SIM, 2011; ZHANG; CHAM, 2012; BAE; DURAND, 2007; TANG; HOU; SONG, 2013; SHEN; HWANG; PEI, 2012; CHEN; CHEN; CHANG, 2016), the patch-based approach (D'ANDRES et al., 2016) and our method for all the 22 images in the database are shown in Table 5.10. Our method outperforms all competitive edge-based methods, being inferior only to (D'ANDRES et al., 2016). In fact, edge-based methods tend to fail when the blurry image does not present many edges, particularly when the local blur changes, which is a natural limitation of any edge-based method. On the other

hand, the method with the best result (D'ANDRES et al., 2016) is based on training a classifier using a discrete set of disc-blur parameters (more precisely, from 0.5 to 5 in steps of 0.25), which leads to a piece-wise constant blur map. Despite the good results achieved by their method, it is prone to failure when the test image contains blur values that were not present in the training step, as acknowledged by the authors.

The blur maps produced by the methods used in the comparative study for all the 22 images of the database are shown in Fig. 5.11 and Fig. 5.12. As it can be observed, our approach presents visually coherent results (and similar to (D'ANDRES et al., 2016)), with low and approximately constant blur values along in-focus areas, and smoothly varying blur values when there are objects lying progressively away from the focal plane. They also present sharp transitions at depth discontinuities, but our method does not produce good results when the image does not present many edges (as in ninth and eleventh rows of Fig. 5.11). In fact, the lack of image edges is a natural limitation of edge-based methods. It is also worth noting that the experiment on this database does not include the proposed refinement step. The results were visually very similar, but the mean absolute error was 0.243.

The average running times for the analyzed methods using all the images in the dataset are also shown in the bottom of Table 5.10. We do not have code or running times for (D'ANDRES et al., 2016) (only the results, kindly provided by the authors), but the regression tree fields used in their method, which is a global image labeling approach, tends to be costlier (particularly for larger images). The proposed approach was only slower than (CHEN; CHEN; CHANG, 2016), but with considerably smaller MAEs. In fact, Table 5.10 indicates that our method presents a very good compromise between MAE and running times when compared to competitive approaches.

Table 5.10: Mean Absolute Errors (MAE) for the images provided in (D'ANDRES et al., 2016).

Image#	Zhuo&Sim	Zhang&Cham	Bae&Durand	Tang et.al.	Shen et.al.	Chen et.al.	Andres et.al.	*OurM.*
01	0.229	0.326	0.397	1.028	0.579	1.023	0.098	0.119
02	0.358	0.399	0.361	0.746	0.510	0.839	0.129	0.172
03	0.233	0.270	0.325	1.866	0.558	0.989	0.106	0.186
04	0.216	0.266	0.554	2.050	0.581	0.938	0.080	0.114
05	0.211	0.268	0.291	1.281	0.644	1.177	0.081	0.160
06	0.210	0.309	0.478	2.444	0.472	0.898	0.073	0.181
07	0.230	0.268	0.581	2.358	0.675	1.056	0.105	0.185
08	0.490	0.465	0.901	4.177	0.419	0.675	0.083	0.364
09	0.404	0.478	0.624	1.239	0.579	0.995	0.069	0.224
10	0.268	0.344	0.445	1.759	0.513	0.951	0.131	0.128
11	0.400	0.412	0.604	2.834	0.566	0.872	0.112	0.190
12	0.432	0.443	0.698	3.056	0.411	0.731	0.077	0.217
13	0.258	0.309	0.381	2.459	0.654	1.103	0.084	0.147
14	0.343	0.380	0.547	1.813	0.683	1.004	0.266	0.264
15	0.535	0.539	0.652	0.388	0.387	0.705	0.076	0.328
16	0.289	0.353	0.416	0.887	0.469	0.926	0.108	0.221
17	0.485	0.522	0.626	2.720	0.405	0.678	0.134	0.289
18	0.324	0.337	0.466	1.525	0.487	0.880	0.105	0.181
19	0.319	0.325	0.514	1.084	0.678	1.020	0.135	0.226
20	0.329	0.397	0.592	1.785	0.523	0.961	0.112	0.158
21	0.296	0.363	0.450	1.063	0.549	0.916	0.094	0.116
22	0.437	0.556	0.687	1.487	0.429	0.880	0.084	0.202
Av. MAE	0.332	0.379	0.527	1.821	0.535	0.919	0.106	0.199
Av. Time(Sec)	9.42	22.58	7.63	11.84	2.72	0.81	*n/a*	1.44

Figure 5.11: Comparison of different blur estimation algorithms using the images(1-12) provided in (D'ANDRES et al., 2016) . Left column: original images. Remaining columns, from left to right: ground truth and results produced by (ZHUO; SIM, 2011), (ZHANG; CHAM, 2012), (BAE; DURAND, 2007), (TANG; HOU; SONG, 2013), (SHEN; HWANG; PEI, 2012), (CHEN; CHEN; CHANG, 2016), (D'ANDRES et al., 2016) and the proposed method, respectively.

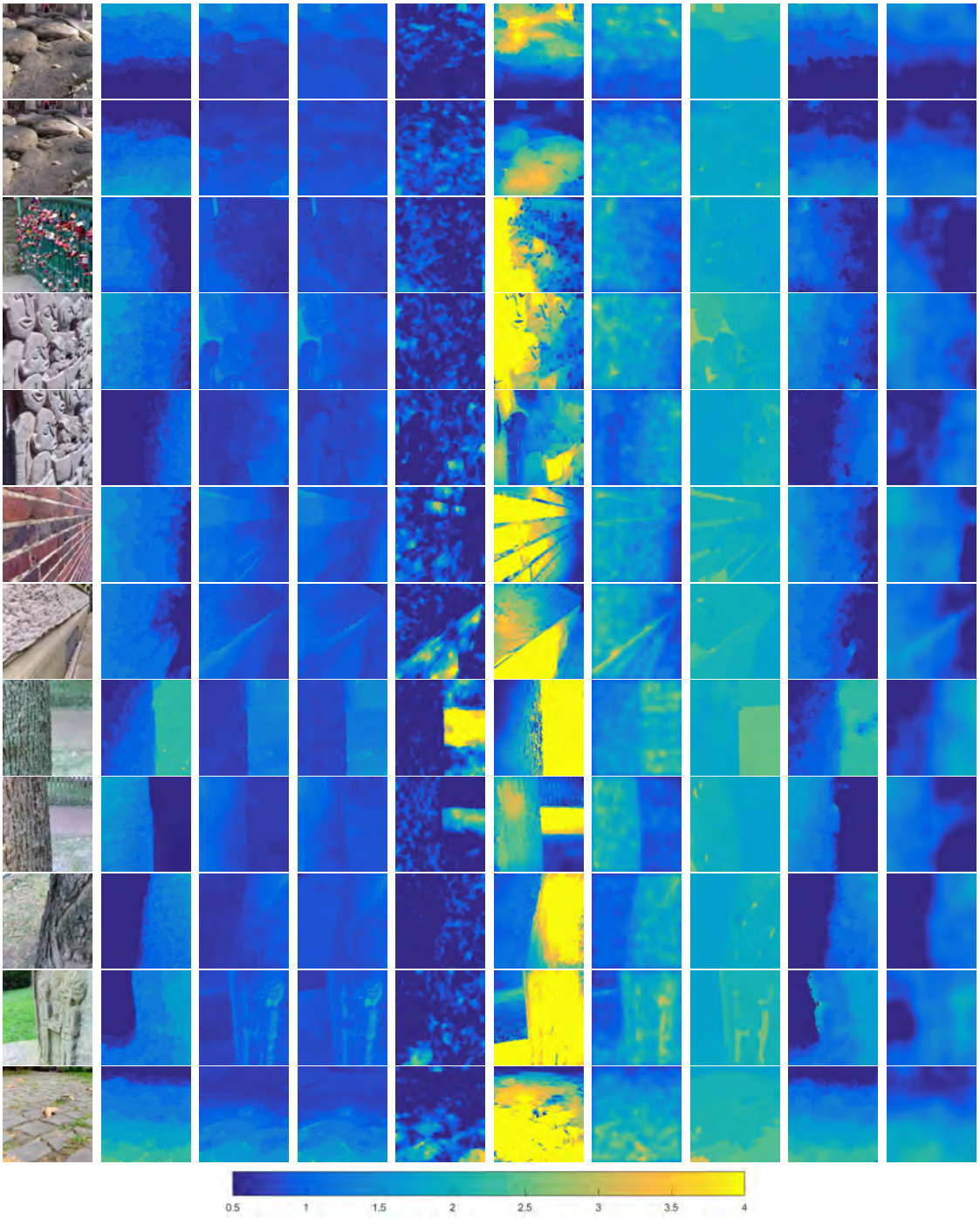
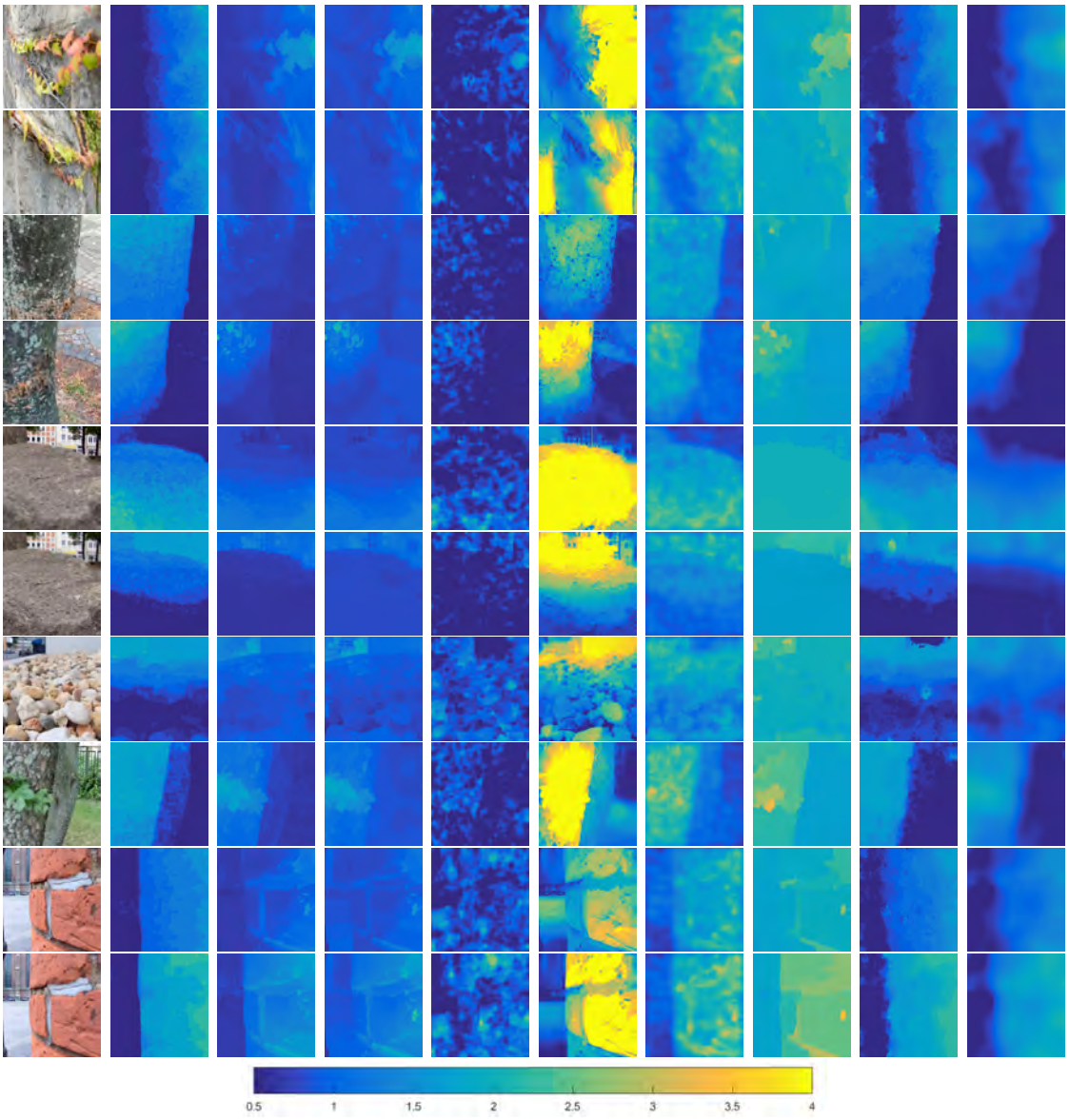


Figure 5.12: Comparison of different blur estimation algorithms using the images(13-22) provided in (D'ANDRES et al., 2016). Left column: original images. Remaining columns, from left to right: ground truth and results produced by (ZHUO; SIM, 2011), (ZHANG; CHAM, 2012), (BAE; DURAND, 2007), (TANG; HOU; SONG, 2013), (SHEN; HWANG; PEI, 2012), (CHEN; CHEN; CHANG, 2016), (D'ANDRES et al., 2016) and the proposed method, respectively.

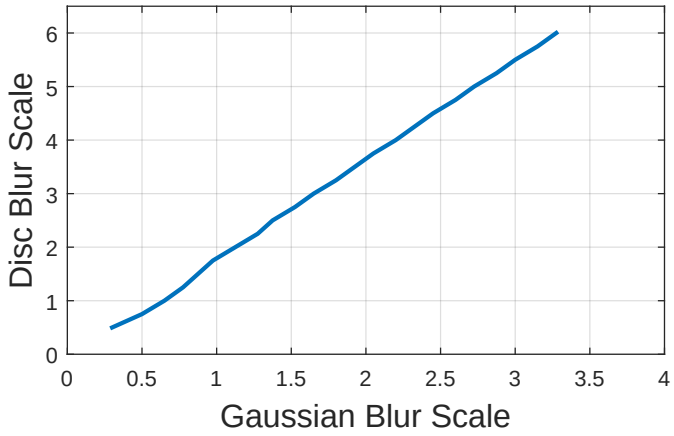


5.3.3 Effect of the Point Spread Function

So far, methods (including our and other edge-based ones such as (ZHUO; SIM, 2011; ZHANG; CHAM, 2012; BAE; DURAND, 2007; TANG; HOU; SONG, 2013; SHEN; HWANG; PEI, 2012; CHEN; CHEN; CHANG, 2016) etc.) that assume naturally blurred images can be simulated with a Gaussian PSF followed a similar formulation in order to compute the unknown blur amount. However, it is very difficult to know the exact defocus blur PSF in natural images.

Recently, a patch-based method (D'ANDRES et al., 2016) is proposed with a defocus blurry image dataset which includes the "known" ground truth blur scales for each image, assuming a disc PSF parametrization. Then, the author computed a conversion table to make a "fair" comparisons with methods that assume Gaussian PSFs, by measuring the closest *Mean Square Error* fits around a blurry patch centered on a single step edge using both disc and Gaussian PSFs. This conversion function can be seen in Fig. 5.13.

Figure 5.13: Disc to Gaussian blur scale conversion function



In order to observe the effect of the PSF, we conducted a set of experiments starting with a synthetic single step edge image, which is artificially blurred by spatially-invariant PSFs for different Gaussian blur levels and corresponding disc levels (computed using the conversion function depicted in Fig. 5.13), and some example images can be seen in Fig. 5.14. Then, we apply Eq. 3.7 using two fixed reblurring parameters[3] (i.e. $\sigma_1 = 1$ and $\sigma_2 = 1.5$) to estimate the blur scales at edge

[3]Since the goal is to observe only the effect of the chosen PSF model, we opted to use the plain formulation to compute blur.

points (sparse blur map) and we observe the Mean Absolute Errors along the edge for each pair of artificially blurred image (Gaussian and corresponding disc blurry).

Figure 5.14: Artificially blurred images. First row using spatially-invariant Gaussian PSFs ($\sigma_b = 0.5$, $\sigma_b = 1.65$ and $\sigma_b = 3$ from left to right, respectively) and second row corresponding disc blurry images ($r_b = 0.75$, $r_b = 3$ and $r_b = 5.5$ from left to right, respectively).

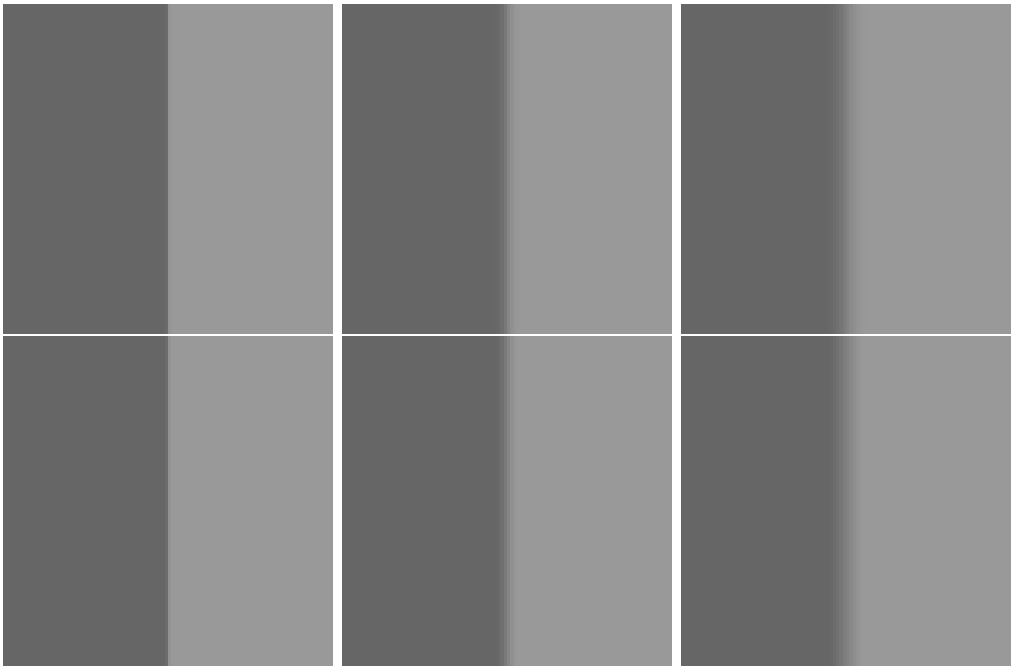


Figure 5.15: Mean Absolute Errors of each artificially blurred image pair (Gaussian and disc blurry) for different Gaussian and disc kernel levels.

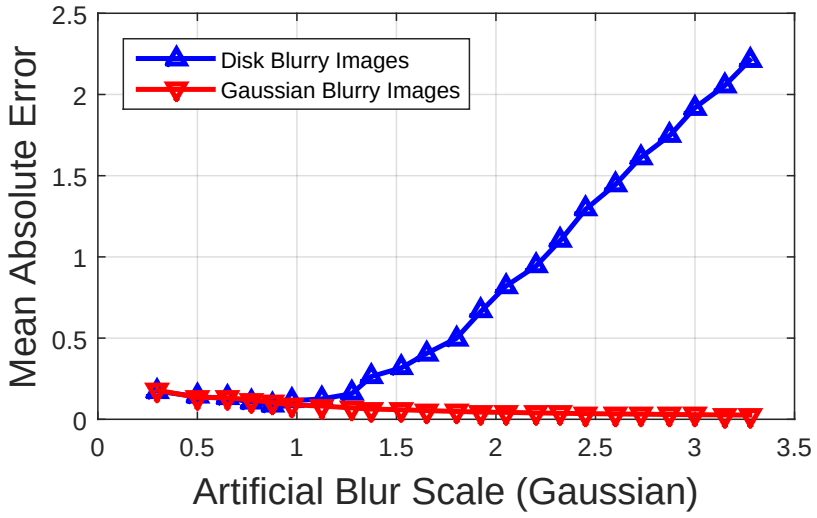


Fig. 5.15 shows the Mean Absolute Errors (vertical axis) of each artificially blurry image pair (Gaussian and disc blurry) as a function of the blur scale (hor-

izontal axes). As it can be observed from Fig. 5.15, the MAE when blurring the image with a Gaussian PSF is small (blue curve). On the other hand, when the image is blurred with a disc PSF, the error produced by our method is larger, and it increases as the input blur amount grows.

A similar experiment is conducted using real images. For that purpose, we chose some natural sharp images that we used before in Subsection 5.3.1 and Subsection 5.3.2. More precisely, the `Brick House` and `Umdan` images from (RUBINSTEIN et al., 2010), and *Image 01* and *Image 19* from (D'ANDRES et al., 2016). These sharp images were artificially blurred by spatially-invariant PSFs for different Gaussian blur levels and corresponding disc levels (which were computed using the conversion function depicted in Fig. 5.13) and some of the results can be seen in Fig. 5.16, Fig. 5.17, Fig. 5.18 and Fig. 5.16, and it can be observed that Gaussian and corresponding disc blurry images are visually very similar.

Figure 5.16: Artificially blurred natural image (*Brick House*). First row using spatially-invariant Gaussian PSFs ($\sigma_b = 0.5$, $\sigma_b = 1.65$ and $\sigma_b = 3$ respectively) and second row corresponding disc blurry images ($r_b = 0.75$, $r_b = 3$ and $r_b = 5.5$ respectively). Source : (RUBINSTEIN et al., 2010)

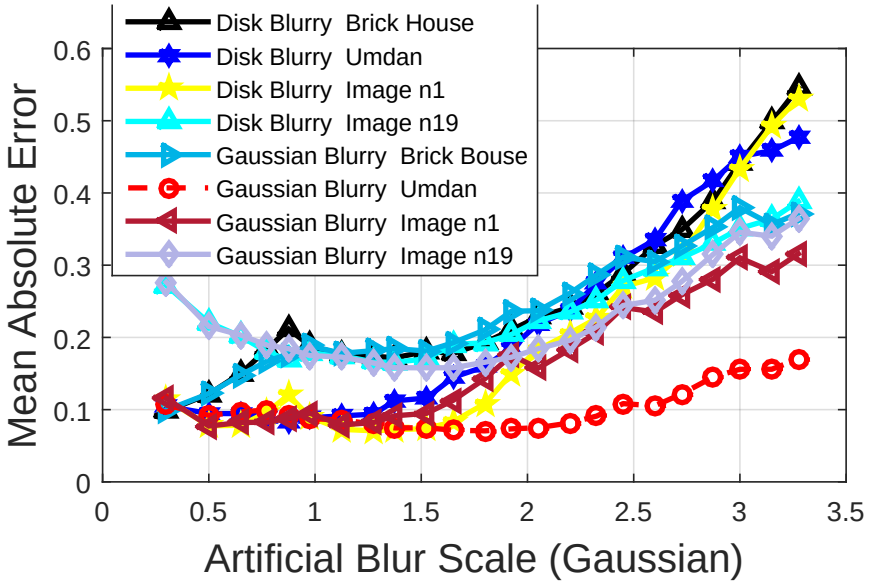
As in the experiment with the synthetic image, the MAE was computed as a function of the blur amount using our 2D multi-scale image gradients method, as shown in Fig. 5.20. Interestingly, the discrepancies when blurring with disc PSF were smaller than those observed for the natural images, particularly for higher blur amounts.

Figure 5.17: Artificially blurred natural image (*Umdan*). First row using spatially-invariant Gaussian PSFs ($\sigma_b = 0.5$, $\sigma_b = 1.65$ and $\sigma_b = 3$ respectively) and second row corresponding disc blurry images ($r_b = 0.75$, $r_b = 3$ and $r_b = 5.5$ respectively). Source : (RUBINSTEIN et al., 2010)

Figure 5.18: Artificially blurred natural image (*Image 01*). First row using spatially-invariant Gaussian PSFs ($\sigma_b = 0.5$, $\sigma_b = 1.65$ and $\sigma_b = 3$ respectively) and second row corresponding disc blurry images ($r_b = 0.75$, $r_b = 3$ and $r_b = 5.5$ respectively). Source : (D'ANDRES et al., 2016)

Figure 5.19: Artificially blurred natural image (*Image 19*). First row using spatially-invariant Gaussian PSFs ($\sigma_b = 0.5$, $\sigma_b = 1.65$ and $\sigma_b = 3$ respectively) and second row corresponding disc blurry images ($r_b = 0.75$, $r_b = 3$ and $r_b = 5.5$ respectively). Source : (D'ANDRES et al., 2016)

Figure 5.20: Mean Absolute Errors of each artificially blurred image pair (Gaussian and disc blurry) for different Gaussian and disc kernel levels.



5.4 Video Defocus Blur Estimation

This section shows the full blur map estimation results for video sequences using our method and also competitive techniques applied in a frame-wise manner. Since it is not to our knowledge the existence of publicly available databases to evaluate defocus blur estimation of video sequences, the experimental validation was performed using real videos extracted from a TV Series called UNDER THE DOME© and a video that we recorded. Results are evaluated only qualitatively through visual inspection of the produced blur map videos due to the lack of ground truth data for video sequences.

The first experiment explores a short video sequence taken by a static camera and without any noticeable object motion, only changes of the focal plane (see some frames in Fig. 5.21). In order to show the estimated blur map results as well as our Kalman filtered version of the blur map sequences, we choose five control points (red points in Fig. 5.21) and evaluate the estimated blur scales at those points along time. More precisely, points #1 and #5 lie on the most blurry regions of the sequence in the beginning, then their blur level reduces. Points #3 and #4 belong to the same object and are initially blurry, becoming in-focus after some time, while point #2 presents an opposite behavior, starting in-focus and then defocused.

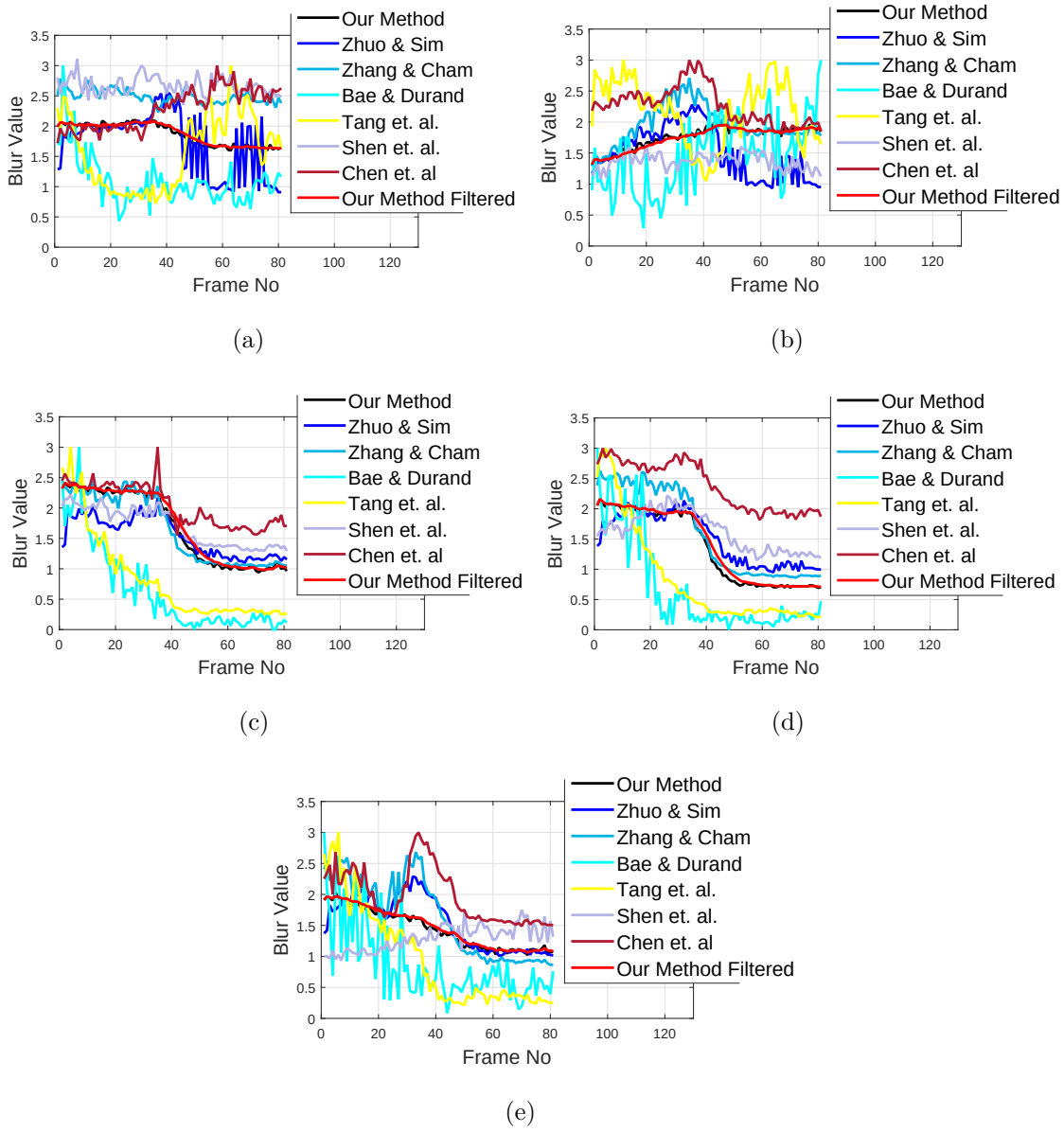
Figure 5.21: Blurry video frames and chosen control points. (a) frame no #1, (b) frame no #32 and (c) frame no #80

(a) (b) (c)

Fig. 5.22 shows the estimated blur values for all five control points obtained with different methods, namely (ZHUO; SIM, 2011), (ZHANG; CHAM, 2012), (BAE; DURAND, 2007), (TANG; HOU; SONG, 2013), (SHEN; HWANG; PEI, 2012) and (CHEN; CHEN; CHANG, 2016), as well our method with and without temporal coherence. Since the spatially/temporally varying blurry video has only focal change along time (frames), the estimated full blur maps should

change smoothly. However, blur estimates in frame-by-frame manner (including our method) creates temporal inconsistencies due to the fact that none of the methods uses temporal information. On the other hand, the proposed temporal scheme presents a smooth temporal variation in time.

Figure 5.22: Estimated blur values for all five control points obtained with different methods as well our method with and without temporal coherence for the video sequence depicted in Fig. 5.21. (a) point #1, (b) point #2, (c) point #3, (d) point #4 and (e) point #5

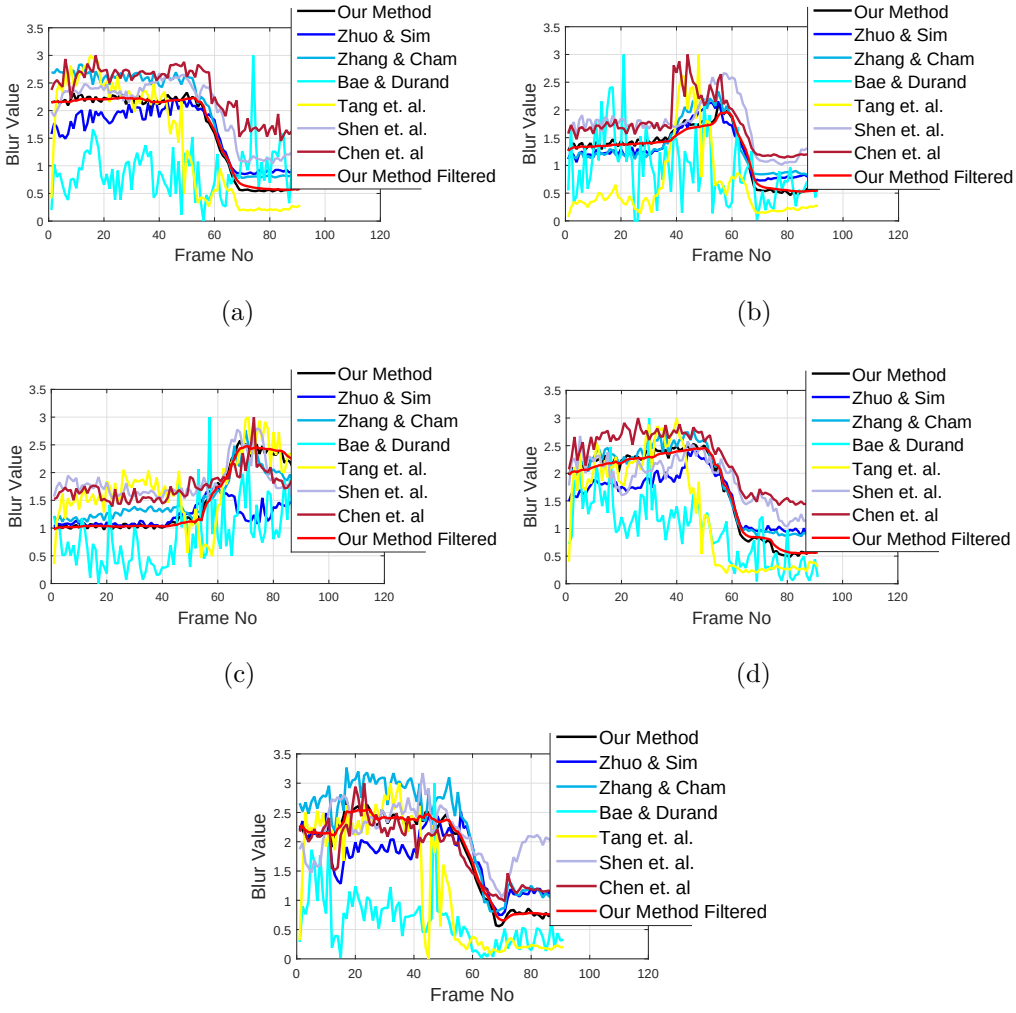


The second experiment is conducted on a more challenging video sequence, which contains moving objects (see three chosen frames in Fig. 5.23). Blur values of some control points (see Fig. 5.23) along time can be seen in Fig. 5.24. Point #1 initially is out-of-focus then turn in-focus. Point #2 relates to an occlusion/disocclusion, starting in-focus, then changing to out-of-focus and finally getting in-focus again. Point #3 is in-focus in the beginning of the video, then becomes out-of-focus; finally, points #4 and #5 belong to out-of-focus regions initially, then become in-focus points. As in the previous experiment, all methods applied in a frame-wise manner introduced temporal inconsistencies between frames, whereas the temporal approach produced smooth trajectories (see blur values of the chosen point along time in Fig. 5.24), while sharp transitions due to occlusions/disocclusions are maintained. However it is worth to note that, even though point #2 is initially in-focus, the obtained blur value has additional blur caused by the light reflection of the hairless head of the actor, which is called 'shading effect' (ELDER; ZUCKER, 1998) and it is a natural limitation for all kind of defocus blur estimation methods.

Figure 5.23: Blurry video frames and chosen control points. (a) frame no #1, (b) frame no #32 and (c) frame no #80

(a) (b) (c)

Figure 5.24: Estimated blur values for all five control points obtained with different methods as well our method with and without temporal coherence for the video sequence depicted in Fig. 5.23. (a) point #1, (b) point #2, (c) point #3, (d) point #4 and (e) point #5



A third (and even more challenging) sequence, which contains object and camera motion, was also used to test the proposed method. Some frames of this sequence are shown in Fig. 5.25, along with control points. The movements (camera or object) create sudden changes between frames, especially for point #1, as shown in Fig. 5.26(a). On the other hand, due to changes in the camera focus, points #2 and #3 become gradually in-focus, while points #4 and #5 present the opposite behavior. As in the previous examples, the Kalman filtered version presents a good compromise between smoothness and transition lag.

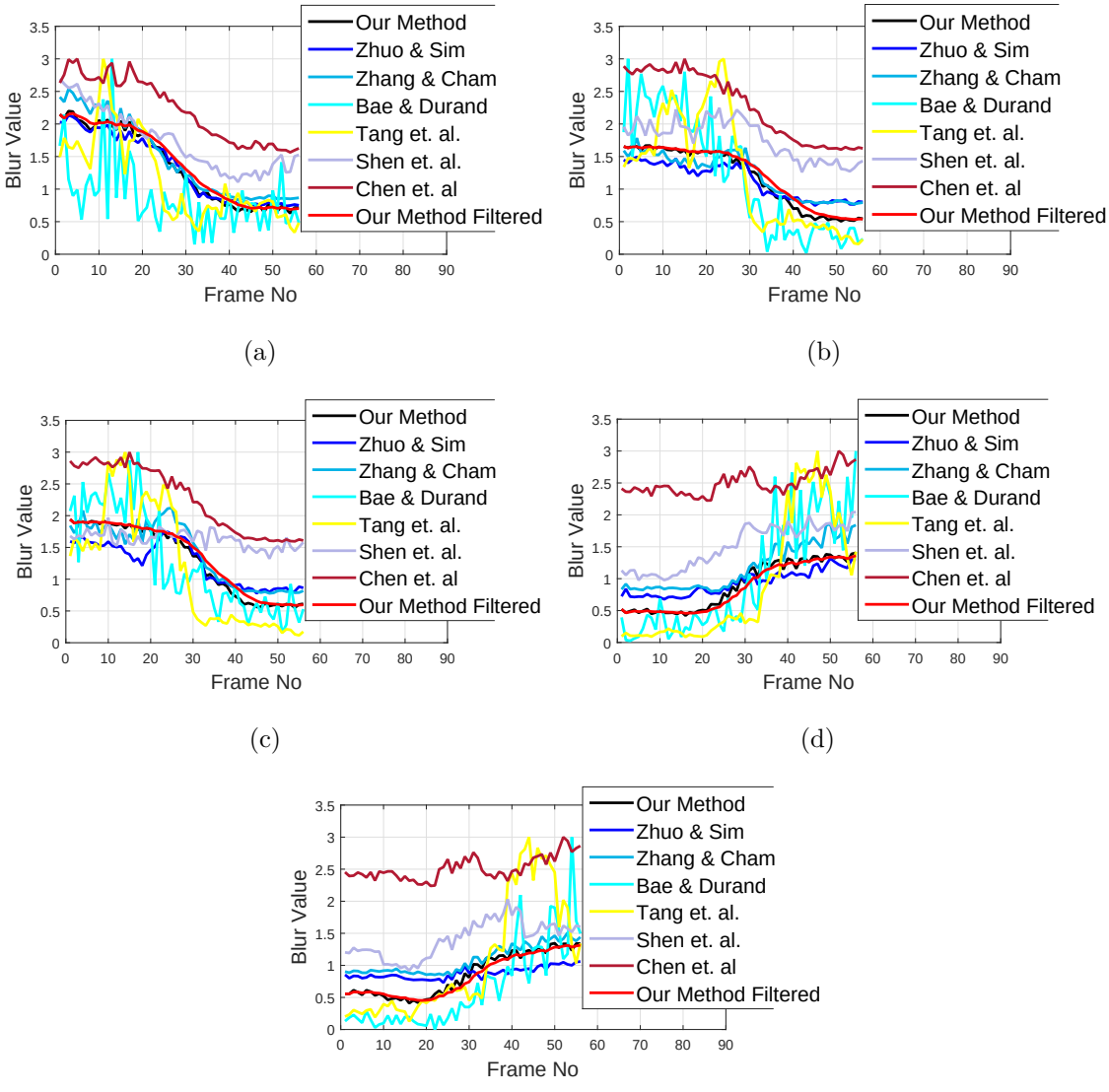
The last experiment relates to a video taken from a DoF camera containing an in-focus object that moves in front of a defocused background, aiming to illustrate how the Kalman filter deals with occlusions/disocclusions. The focal plane of the camera was kept fixed during the sequence, so that the defocus level of each object is still due to their distance to the camera. As in the previous experiments, some frames along with control points were selected, as shown in Fig. 5.27. Control point #1 alternates from in-focus and defocused regions, since it belongs to different regions in time: hand, background, face and background again. Point #2 is initially in-focus, placed on the hat of the doll, then becomes out-of-focus. The plots of the estimated defocus blur using different methods are shown in Figs. 5.27(a) and 5.27(c), while Figs. 5.27(b) and 5.27(d) focus on the comparison of the proposed approach with and without temporal coherence. It can be observed that the Kalman filtered provides good compromise between smoothness and transition lag, especially around frames 45 and 85 for point #1. On the other hand, point #2 is constantly on a blurry region after doing a sharp transition from the hat of the doll around frame 10.

Although the plots shown in this section indicate that the temporal approach reduces jittering, a better visual analysis can be performed by evaluating the processed video sequences. For that, please refer our web-page <https://sites.google.com/site/axkaraali/dissertationvideos>.

Figure 5.25: Blurry video frames and chosen important points. (a) frame no #1, (b) frame no #32 and (c) frame no #80

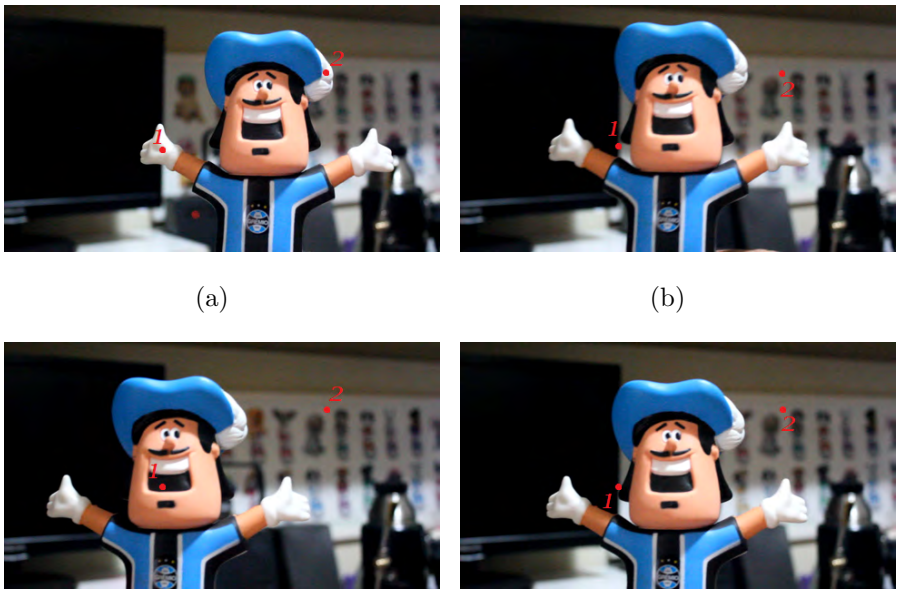
(a) (b) (c)

Figure 5.26: Estimated blur values for all five control points obtained with different methods as well our method with and without temporal coherence for the video sequence depicted in Fig. 5.25. (a) point #1, (b) point #2, (c) point #3, (d) point #4 and (e) point #5



(a) (b) (c) (d) (e)

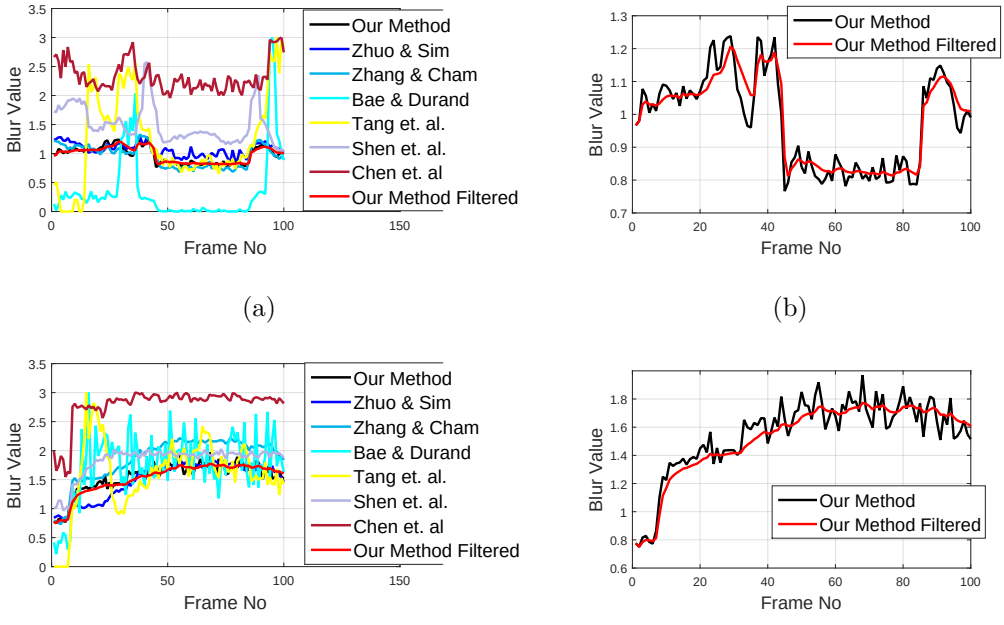
Figure 5.27: Blurry video frames and chosen control points. (a) frame no #1, (b) frame no #31, (c) frame no #50 and (d) frame no #100



(a) (b)

(c) (d)

Figure 5.28: Estimated blur values for two control points obtained with different methods as well our method with and without temporal coherence for the video sequence depicted in Fig. 5.25. (a)-(c) point #1 and point #2 for all methods, (b)-(d) point #1 and point #2 for our method with and without temporal coherence



(a) (b)

(c) (d)

5.5 Blur-aware Image Retargeting

Existing cameras and display devices present a wide range of resolutions and aspect ratios. Such diversity introduced a new problem: content-aware image retargeting, which aims to adjust the content of a digital image to a specific display, maintaining a pleasant visual quality and keeping relevant objects of the image. Clearly, one key issue in content-aware image retargeting is to identify the relevant components in a photograph, which is very application dependent. Different types of importance measures were proposed in the literature, and they are used to determine which pixels or regions will be removed, kept or modified. Therefore, the first step of a content-aware retargeting application is to produce an importance map, which typically includes low and/or high level feature extraction, and occasionally some additional constraints such as line/structure/object detection. Then, a retargeting operator itself is applied to the image combining the importance map and additional constraints (if any) in order to create the final image.

Image retargeting is a relatively new field in the area of computational photography, but various algorithms have been proposed for this problem, including comprehensive reviews of recent methods (VAQUERO et al., 2010; RUBINSTEIN et al., 2010). In a nutshell, existing methods can be classified basically into two categories: discrete and continuous methods.

Discrete image retargeting methods typically remove pixels (or patches) while preserving the *important* parts of the image, and content-aware cropping can be given as one of the earliest example of these methods. The work of Suh et al. (SUH et al., 2003), which provides an algorithm that crops images before generating a thumbnail image automatically, can be cited as one of the pioneers. This algorithm calculates a saliency map and seeks a cropping window that maximizes the percentage of salient points within the window, using a greedy approach. Another typical example of a discrete retargeting operator is seam carving (AVIDAN; SHAMIR, 2007). The idea behind seam carving is to iteratively remove continuous pixel chains, namely seams, while preserving the content as much as possible. Countless improvements were proposed over the original seam carving paper during the last two decades. One of the most notable ones was proposed by Rubinstein et al. (RUBINSTEIN; SHAMIR; AVIDAN, 2008), which suggests a new approach to handle the fact that seam removal may add more energy into the energy map. For

that purpose, a new forward energy function that measures the effect of seam carving on the retarget image was introduced. Continuous methods for retargeting are based on warping, which deals mostly with images. Liu and Gleicher (LIU; GLEICHER, 2005), for example, proposed to use a fisheye-view warping function in order to amplify the region-of-interest (ROI), while de-emphasizing less important regions. However, the method assumes that the image has only one ROI, meaning that it cannot handle images with multiple objects.

As it can be observed, the main obstacle of retargeting algorithms is to choose the most important parts (according the observers) in the image to create visually plausible results. Existing algorithms try to handle the difficulty of choosing the right parts of the image using some kind of additional information, such as saliency detectors, faces, edge maps, etc. Blur information has been used for saliency detection before (JIANG et al., 2013), but it is not to our knowledge its use in the context of image retargeting. Hence, we propose a retargeting scheme that accounts for blur information as an additional contribution to this dissertation, which was published in (KARAALI; JUNG, 2016).

Let us consider an input image with dimensions $R \times C$, and the desired dimension of the output image is $r \times c$, with $r \leq R$ and $c \leq C$. The core of our approach is to identify and preserve in-focus regions in a photograph. For that purpose, we initially compute a dense blur map of the image, and classify as in-focus regions those that present low blur. Then, we remove the boundaries of the image taking into account the in-focus regions, since cropping tends to produce good visual results (MA et al., 2012). If just cropping is not enough to achieve the desired image dimension, seam carving is applied using a novel blur-aware energy function.

5.5.1 Blur-Aware Image Cropping

In-focus objects clearly are those that present low blur values, so that they can be detected by using a threshold T_σ. Even though "acceptable" sharpness depends on several aspects (such as image resolution, size of the objects of interest, viewing resolution, etc), which impacts the choice of the threshold, we used a fixed value $T_\sigma = 1$ as suggested in (ZHUO; SIM, 2011) in order to find in-focus regions in our defocus blur map.

Since the blur estimation process may contain errors (particularly in homo-

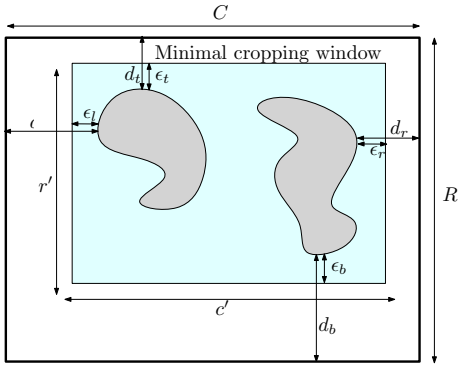
geneous textureless areas), the blur value of in-focus objects may vary. As a consequence, the thresholding process may separate one object into two or more. To alleviate this problem, we apply a morphological opening operator on top of the full blur map $B(x, y)$, obtaining $B'(x, y) = s(x, y) \cdot B(x, y) = (s(x, y) \ominus B(x, y)) \oplus s(x, y)$, where $\oplus$ and $\ominus$ denote dilation and erosion, respectively, using a circular structuring element $s(x, y)$ with dimension $N_s \times N_s$, where $N_s = 0.015 \min\{R, C\}$ (i.e., a fraction of the smallest dimension of the input image). In fact, the morphological opening tends to connect small gaps between darker regions, which relates to propagating in-focus regions in our context.

Even after the morphological opening, the blur-based thresholding procedure may produce several small regions, some of them probably irrelevant. Our criterion to select relevant in-focus regions in based on the area of the objects. After binarizing the blur map, we identify the N connected components and compute the largest area, called A_{max}. Then, a given connected component is kept if its area A is larger than a fraction $0 < T_A < 1$ of the largest object. In this work, we used $T_A = 0.5$, so that objects less than 50% of the largest are discarded.

Given the objects of interest, we compute the minimum distance of the bounding box containing these objects to the left, right, top and bottom image boundaries. Let d_l, d_r, d_t and d_b denote such distances, and let ϵ_l, ϵ_r, ϵ_t and ϵ_b denote the minimum desired distances from the objects of interest in the cropped image (so that ideally no object touches the boundaries of the retargeted image).

The goal of image cropping is to find a rectangular cropping region $\mathcal{R}$ with dimensions $r' \times c'$, with $r' = \max\{r, R - d_t - d_b + \epsilon_t + \epsilon_b\}$ and $c' = \max\{c, C - d_l - d_r + \epsilon_l + \epsilon_r\}$, so that it is able to keep all the objects of interest, and also the boundary tolerances ϵ_i, as shown in Fig. 5.29. If the target image dimensions r and c are large

Figure 5.29: Illustration of the minimal cropping region.



enough, there are several possible cropping windows that satisfy the desired criteria (i.e. keep all objects with some boundary tolerance). Since our goal is to prioritize in-focus regions, the optimal crop is based on the preservation of low-blur pixels. More precisely, if $\mathcal{R}(x, y)$ denotes a rectangular region with dimensions $r' \times c'$ which top-right position is given by (x, y), the location (x^*, y^*) of the optimal cropping window is given by

$$(x^*, y^*) = \arg min_{(x,y)} \sum_{(m,n)\in\mathcal{R}(x,y)} B'(m, n), \tag{5.1}$$

where B' is the processed blur map. In our implementations, we make use of integral images (or summed area tables), so that each summation over a rectangular region in Eq. 5.1 is performed at constant time.

5.5.2 Image Retargeting With Seam Removal

If $r = r'$ and $c = c'$, only cropping is enough to produce the retargeted image at the desired resolution. Otherwise, an additional step based on seam carving (RUBINSTEIN; SHAMIR; AVIDAN, 2008) is applied. However, instead of using the traditional gradient-based energy to guide the seam removal process, we propose a new strategy that takes the defocus blur of the images also into account in order to compute the importance map, aiming to preserve in-focus regions. As such, the new energy function should present higher costs at in-focus regions (i.e. low blur regions) to prevent seams from crossing them.

Given the morphologically processed blur map $B'(x, y)$, we build a "sharpness" map E_b using a sigmoid transducer function inspired on (VU; PHAN; CHANDLER, 2012):

$$E_b(x, y) = 1 - \frac{1}{1 + e^{\alpha_1(B'(x,y)-\alpha_2)}}, \tag{5.2}$$

where α_1 and α_2 are the constants that control the shape of the transducer. Experimentally, we have set $\alpha_1 = -4$ and $\alpha_2 = 2$, which generates a monotonically decreasing function that is roughly flat for small blur values B' and decays smoothly.

The blur energy E_b is then combined with a directional gradient-based energy, and the seam extraction process is applied using the forward energy approach similarly to the Improved Seam Carving (ISC) method (RUBINSTEIN; SHAMIR;

AVIDAN, 2008), in which the optimal seam is computed using a dynamic programming approach using an 8-connectedness neighborhood for each pixel looking for a seam which removal inserts the minimum amount of energy into the retargeted image. In such approach, the gradient-based cost C_T is computed differently for paths coming from the left, up or right of the pixel under analysis (assuming a vertical seam, from top to bottom):

$$\begin{aligned} C_L(x,y) &= |I(x,y+1) - I(x,y-1)| \\ &\quad + |I(x-1,y) - I(x,y-1)| \\ C_U(x,y) &= |I(x,y+1) - I(x,y-1)|, \\ C_R(x,y) &= |I(x,y+1) - I(x,y-1)| \\ &\quad + |I(x-1,y) - I(x,y+1)| \end{aligned} \tag{5.3}$$

The proposed energy function $E = \min(E_1, E_2, E_3)$ is then computed based on the minimum energy from the three possible directions, where E_1, E_2 and E_3 are combinations of gradient and blur information given by

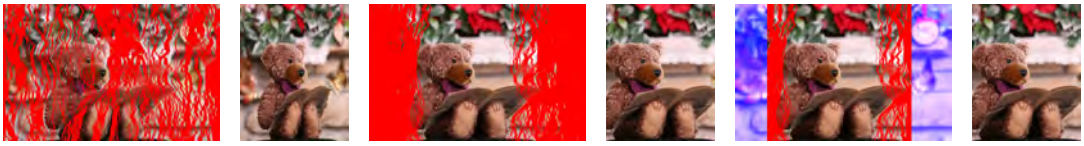
$$\begin{aligned} E_1 &= E(x-1,y-1) + ((C_L(x,y)+\delta_1)(E_b(x,y)+\delta_2)), \\ E_2 &= E(x-1,y) + ((C_U(x,y)+\delta_1)(E_b(x,y)+\delta_2)), \\ E_3 &= E(x-1,y+1) + ((C_R(x,y)+\delta_1)(E_b(x,y)+\delta_2)) \end{aligned} \quad , \tag{5.4}$$

so that in-focus regions with sharp edges tend to be avoided (i.e., present larger energy values). The offset parameters δ_1 and δ_2 control the importance of the blur and gradient energy maps in the combined map: If $\delta_1 = 0$, a region without significant texture ($C_k \approx 0$) will cause the final energy E to be very close to zero regardless of the defocus blur degree; similarly, if $\delta_2 = 0$, a region with significant blur ($E_b \approx 0$) will cause the same effect, regardless of the gradient term. We have noticed in our experiments that such setting tends to produce noticeable seams in the retargeted image, since they are concentrated in textureless or very blurred regions of the image. Small values for δ_1 and δ_2 can avoid these situations, and we have noticed experimentally that $\delta_1 = 0.0001$ and $\delta_2 = 0.01$ present a good compromise between seamless retargeting and in-focus object preservation.

5.5.3 Experimental Results of Blur-Aware Retargeting

For the experimental results, we used some of the images explored in (ZHUO; SIM, 2011), some images from a TV Series, and some images from the RetargetMe database (RUBINSTEIN et al., 2010), which also presents the results from other retargeting methods with a given reduction ratio. In all experiments, we used $\epsilon_l = \epsilon_r = \epsilon_t = \epsilon_b = 10$ pixels as the boundary thresholds.

Figure 5.30: (a)-(c) Seams using (RUBINSTEIN; SHAMIR; AVIDAN, 2008) and our energy, respectively. (b)-(d) Retargeted images related to (a)-(c). (e) Cropping + seams using our complete approach, and (f) our final retargeting result.

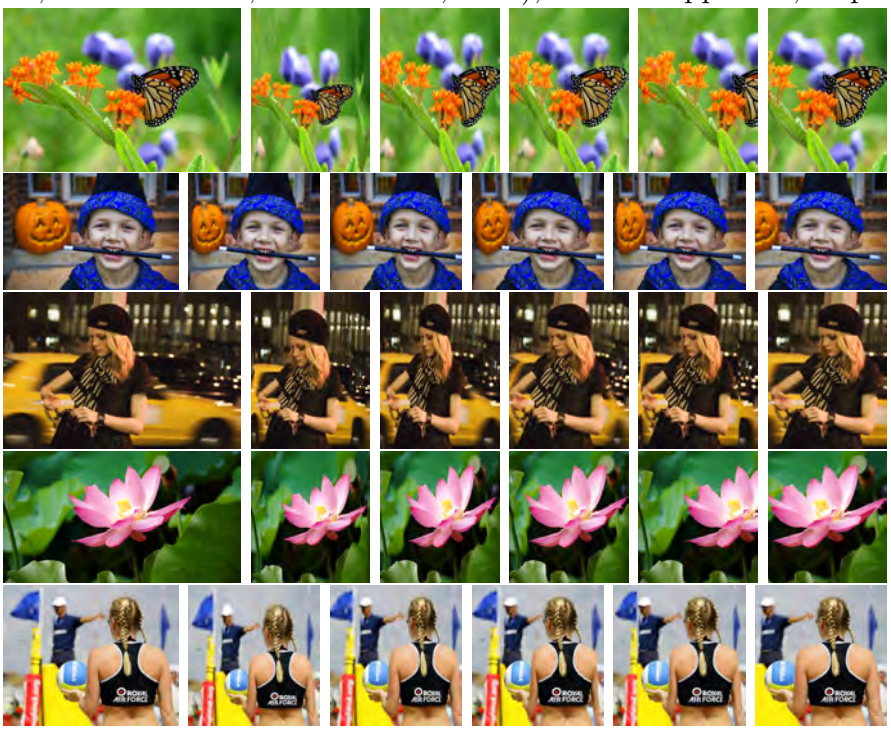


(a) (b) (c) (d) (e) (f)

To evaluate the importance of the blur term, we compared the seams produced by the gradient-based energy (RUBINSTEIN; SHAMIR; AVIDAN, 2008) and our gradient+blur energy. The underlying seam carving algorithm is the same, only the energy function changed (to evaluate just the produced seams with and without blur, cropping was disabled in this example). The results are shown in Fig. 5.30(a) and Fig. 5.30(c), respectively, while the corresponding retargeted images are shown in Fig. 5.30(b) and Fig. 5.30(d), respectively. As it can be observed, the seams produced by our method mostly avoid the in-focus object, which is well preserved in the corresponding retargeted image. The gradient-based method, on the other hand, produces several seams along the teddy bear, which generates a deformed retargeted image. We also illustrate the result of our full retargeting approach combining cropping and seam carving in Fig. 5.30(e), with the final retargeted image shown in Fig. 5.30(f). It can be noticed that just a few additional seams must be removed to reach the desired resolution with the inclusion of cropping, which reduces the computational cost (since cropping is much faster than multiple seam removals).

To compare the proposed approach with competitive retargeting methods, we chose a subset of images acquired with shallow DoF (Depth-of-Field) from (RUBINSTEIN et al., 2010) (recall that the scope of this paper is to deal with defocus blur). Fig. 5.31 illustrates, in the first column, the original images, `Butterfly`,

Figure 5.31: First column: images from *RetargetMe* database (RUBINSTEIN et al., 2010). Second to sixth columns: results of Optimized Scale and Stretch (WANG et al., 2008), Multi-operator (RUBINSTEIN; SHAMIR; AVIDAN, 2009), Improved Seam Carving (RUBINSTEIN; SHAMIR; AVIDAN, 2008), Nonhomogeneous warping (WOLF; GUTTMANN; COHEN-OR, 2007), and our approach, respectively



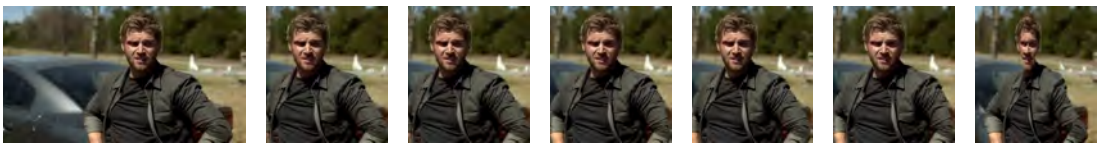
`Child`, `DKNY girl`, `Lotus` and `Volleyball`. The second, third, fourth, fifth and sixth columns show, respectively, the results produced by Optimized Scale and Stretch (OSS) (WANG et al., 2008), Multi-operator (MULTIOP) (RUBINSTEIN; SHAMIR; AVIDAN, 2009), Improved Seam Carving (ISC) (RUBINSTEIN; SHAMIR; AVIDAN, 2008), Nonhomogeneous warping (WARP) (WOLF; GUTTMANN; COHEN-OR, 2007) and our approach.

For the `butterfly` image, the desired dimension requires both cropping and blur-aware seam carving. The butterfly (the in-focus object) was best preserved by using our approach, whereas OSS and WARP clearly deformed the butterfly, MULTIOP and ISC kept irrelevant background while changing the aspect ratio of the butterfly with unnatural leaf deformation. This deformation is also noticeable for the `DKNY girl` image, since OSS operator deforms the left arm while making the head looks narrower, MULTIOP and ISC shrink the women in a very unpleasant way, making her very skinny, and WARP deforms the right arm and head of the woman. On the other hand, our retargeting operator is able to successfully keep the aspect ratio of the girl. Similarly, only cropping is not enough to keep the flower intact in the `Lotus` image. For this image, OSS and MULTIOP produced a

good visual result at keeping the flower (although with a little bit of shrinking), but the ISC operator deformed the middle part of the flower a little bit (especially the middle leaf of the flower), and the WARP operator completely cut the flower (which is the most important object in the image). Whereas, our method was able to keep the size of the flower without any artifacts.

Despite half of the pumpkin was removed by our method in the `Child` image, the kid (only in-focus-object) was best preserved intact. The pumpkin was better preserved by both OSS, MULTIOP, ISC, and WARP but at the cost of deforming the kid (the face was narrowed, and the baton shrunk). Finally, for `Volleyball` image, the deformation can be seen clearly on OSS, ISC and WARP's results, since the arms were skinned with respect to other body parts and the MULTIOP operator shrinks the woman in a very unpleasant way. However, even though the flag on the left disappears, our method yields the best visual result while keeping the woman natural, which is the only in-focus object.

Figure 5.32: (a) The original image (taken from Under The Dome TV Series). (b)-(f) Final retargeting results using our blur aware approach (cropping + seams) with different defocus blur estimation methods, namely (ZHANG; CHAM, 2012), (BAE; DURAND, 2007), (TANG; HOU; SONG, 2013), (ZHUO; SIM, 2011) and our method respectively. (g) Retargeting result without blur estimation, using (RUBINSTEIN; SHAMIR; AVIDAN, 2008).



(a) (b) (c) (d) (e) (f) (g)

Finally, we conducted an experiment to explore the impact of the defocus blur estimation method on our blur aware retargeting approach. For that purpose, we applied our retargeting method to a shallow DoF image using different defocus blur map estimation algorithms, namely (ZHANG; CHAM, 2012), (BAE; DURAND, 2007), (TANG; HOU; SONG, 2013), (ZHUO; SIM, 2011) and our method shown in Fig. 5.32(b) to Fig. 5.32(f). For the chosen default blur estimation technique, we used $T_\sigma = 1$ as the in-focus threshold, as described before. For the other methods, we adjusted the threshold manually (since the blur ranges may vary). As it can be observed, the proposed approach produces plausible results for other blur estimation methods as well. For the sake of comparison, we also showed the re-

sult of gradient-based seam carving (RUBINSTEIN; SHAMIR; AVIDAN, 2008) in Fig. 5.32(g). Clearly, the use of blur information presents potential to improve the visual quality of the retargeted image.

5.6 Image and Video Deblurring

It is important to point out that in many scenarios the quality of a given algorithm is highly dependent on the final application. For instance, the classical stereo matching problem can be used as input for view interpolation. In that context, it was shown that the best methods for stereo matching considered popular objective metrics (such as the percentage of "bad pixels" used in the Middlebury dataset (SCHARSTEIN; SZELISKI, 2002)) might not produce the best interpolated view according to objective image quality metrics PSNR or SSIM (FüHR et al., 2013). Analogously, blur estimation can be viewed as an intermediate step required in non-blind image reblurring or deblurring. In this dissertation, we also evaluate the quality of the proposed blur estimation method, as well as competitive approaches, by measuring the PSNR of deblurred images. This strategy was actually used in (D'ANDRES et al., 2016) for still images, but they evaluated naturally blurred images with artificial noise contamination, so that the final comparison involves both deblurring and denoising.

Additionally, we also perform a correlation analysis between the blur estimation error (MAE) and the PSNR of the corresponding deblurred image. Although a negative correlation might be expected, that is not always the case, as shown in next section.

5.6.1 Image Deblurring

When using non-blind deconvolution methods, the quality of the deblurred images depend on the estimated blur map and also on the deblurring algorithm itself. In this dissertation, we used the combination of deblurring methods proposed in (KRISHNAN; FERGUS, 2009) and (LEVIN et al., 2007) fed by the proposed full blur maps to evaluate the quality of the deblurred images produced using our maps.

More precisely, the deblurring model is given by

$$\hat{I} = \arg\min_{I}(\delta_{data}(I) + \lambda\delta_{reg}(I)), \tag{5.5}$$

where $\hat{I}$ is an estimate of the original (sharp) image I, $\delta_{data}(I)$ is a data-fidelity term, and $\delta_{reg}(I)$ is the regularization term. More precisely, $\delta_{data}(I) = \|h \otimes I - I_b\|_2^2$, where h is the spatially varying (known) blur kernel, and I_b is the observed blurry image. For the regularization term, we use sparse derivative priors $\delta_{reg}(I) = \|f_1 \otimes I\|_\alpha + \|f_2 \otimes I\|_\alpha$, where f_1 and f_2 are first order derivatives in the horizontal and vertical directions, respectively, and $\|.\|_\alpha$ is the *alpha-norm* that acts as a penalty function, set to 0.8 based on hyper-Laplacian priors (KRISHNAN; FERGUS, 2009). We set $\lambda = 0.002$ based on experiments, aiming to maximize the PSNR of the deblurred image. A very similar deblurring algorithm was also employed in (D'ANDRES et al., 2016), but they also consider a blur map estimation error E_h in the objective function. We opted not to use such term to perform a fair comparison of different blur estimation methods in the context of image deblurring, although the use of such term can indeed improve the results.

We have applied the deblurring algorithm to our blur maps, as well as to the ground truth data provided in (D'ANDRES et al., 2016), and the defocus maps produced by competitive blur estimation methods. The deblurred images were then compared with the original all-in-focus image, allowing an objective comparison. This process was performed for the same 22 images used in the previous test, and the resulting PSNRs are summarized in Table 5.11. It can be observed that only (D'ANDRES et al., 2016) and our method presented a higher average PSNR value than the original blurry images, along with the results produced by the ground truth blur values. For the sake of fairness, it is important to point out that deblurring results with disc kernel formulation reported in (D'ANDRES et al., 2016) using the additional error correction term for estimated blur maps (as mentioned before) for artificially noisy versions of the same dataset yielded PNSR gains of 2.06dB ($\sigma_n = 1$) and 1.55dB ($\sigma_n = 2.55$), compared to 0.58dB gain and 1.05dB gain with error correction term when Gaussian formulation is used for the noiseless version of the dataset. In fact, using the error correction term boosts the PSNR of deblurred images using all blur estimation methods, as shown in the bottom of Table 5.11. In both cases, the average PSNR gains of 0.92dB and 1.42dB (with error correction

Table 5.11: PNSRs for the deblurred images using the data provided in (D'ANDRES et al., 2016)

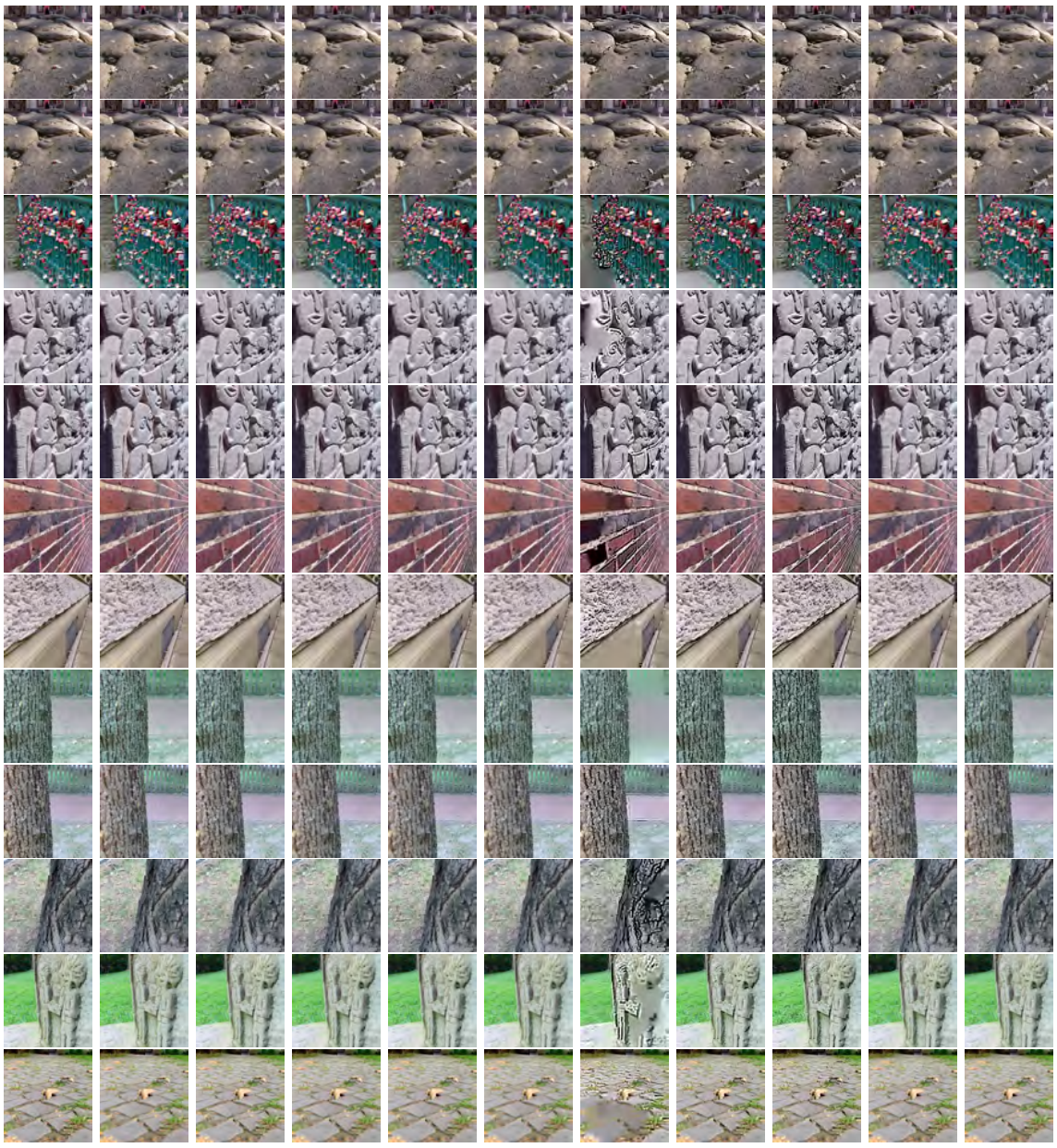
Image#	Blurry	GT Data	Zhuo&Sim	Zhang&Cham	Bae&Durand	Tang et.al.	Shen et.al.	Chen et.al.	Andres et.al.	*OurM.*
01	25.51	26.48	24.90	23.77	25.31	21.14	22.35	19.75	26.25	26.07
02	24.54	25.19	24.18	23.79	24.22	23.62	22.53	19.80	24.90	24.89
03	23.74	25.23	23.60	22.88	23.51	16.35	20.12	16.75	24.76	24.21
04	23.58	25.98	24.87	24.61	24.10	16.17	21.12	17.74	25.56	25.64
05	26.50	28.32	25.17	23.85	25.82	17.35	20.28	16.65	27.83	26.81
06	24.05	25.06	21.47	19.95	23.30	15.52	21.04	15.92	23.87	24.66
07	27.09	28.59	26.02	25.70	26.37	17.98	22.32	18.87	28.28	27.49
08	23.47	24.48	23.95	23.97	23.23	19.82	21.61	18.30	24.40	23.92
09	22.56	23.35	21.94	21.23	22.41	20.00	21.49	19.13	23.26	22.70
10	23.97	25.13	24.09	22.14	24.25	15.92	20.89	16.63	24.93	24.87
11	23.47	24.04	23.37	23.11	23.43	17.12	23.04	21.63	23.87	23.81
12	23.73	24.49	23.70	23.62	23.65	19.24	23.97	21.95	24.36	23.96
13	28.09	28.41	24.95	23.25	27.17	19.02	21.80	17.80	28.13	28.40
14	24.75	25.17	24.52	24.05	24.70	17.92	21.77	19.32	25.47	25.19
15	20.43	21.17	19.84	19.04	20.32	20.57	19.33	16.66	21.08	20.48
16	23.87	25.14	24.16	21.64	24.26	19.70	19.95	15.28	25.01	24.60
17	24.33	24.10	22.40	21.60	23.01	18.84	21.76	17.69	22.69	23.75
18	22.71	23.73	22.74	22.40	22.70	18.24	22.82	20.99	23.53	23.17
19	24.31	25.47	24.11	24.22	23.84	20.81	22.23	19.38	25.11	24.61
20	22.23	22.63	21.46	20.63	21.91	17.13	20.94	17.86	21.92	22.38
21	25.63	26.47	25.26	24.36	24.76	23.23	23.61	20.52	26.08	26.23
22	23.91	24.24	23.24	22.50	23.04	21.54	22.90	20.41	24.06	23.95
Avg. PSNR	24.20	25.13	23.63	22.83	23.88	18.96	21.72	18.59	24.79	24.63
Avg. Gain	*n/a*	0.92	−0.56	−1.37	−0.32	−5.23	−2.48	−5.60	0.58	0.42
Average results with the blur error correction term added to Eq. (5.5)										
Avg. PSNR	24.20	25.63	23.80	23.04	24.08	19.17	21.75	19.36	25.26	24.93
Avg. Gain	*n/a*	1.42	−0.39	−1.16	−0.12	−5.03	−2.45	−4.83	1.05	0.72

term) obtained using ground truth blur maps can be considered an upper bound for deblurring, at least for the chosen deblurring algorithm, and the chosen parameters (clearly, different results will be reached if the deblurring method or its parameters are changed).

Fig. 5.33 and Fig. 5.34 show some deblurring examples using the ground truth blur map data and the defocus blur maps obtained with the estimation methods analyzed in Table 5.11. Although visual analysis is very subjective, it can be observed that only deblurring results fed by our blur map and (D'ANDRES et al., 2016) are close to the original sharp image. On the other hand, other methods do not seem to produce visually good results. For instance, instead (SHEN; HWANG; PEI, 2012) produces disturbing ringing artifacts (seventh column of Fig. 5.33 and Fig. 5.34).

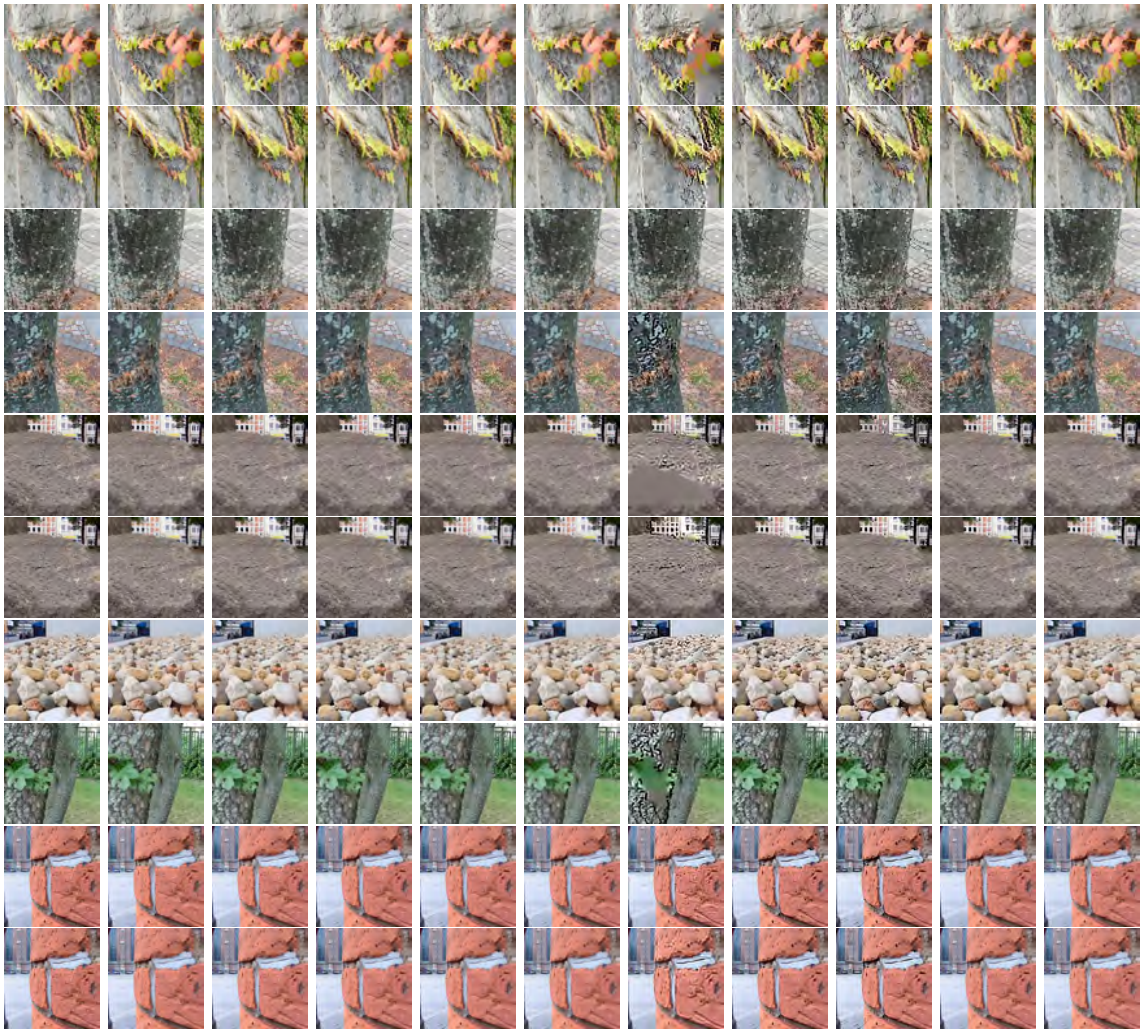
Still in the context of non-blind image deblurring, it would be very interesting to evaluate how a given objective error measure (e.g. MAE) used to compare defocus blur estimation methods relate to an objective error measure (e.g. PNSR) used

Figure 5.33: Comparison of deblurring results using different blur maps using the images(1-12) provided in (D'ANDRES et al., 2016). Left column: original sharp image, blurry image. Remaining columns, from left to right: deblurring result using ground truth data and blur map of (ZHUO; SIM, 2011), (ZHANG; CHAM, 2012), (BAE; DURAND, 2007), (TANG; HOU; SONG, 2013), (SHEN; HWANG; PEI, 2012), (CHEN; CHEN; CHANG, 2016), (D'ANDRES et al., 2016) and the proposed method, respectively. (Best view can be seen zooming)



to compare deblurring/restoration algorithms. Although important, it is not to our knowledge the existence of such analysis, probably due to the lack of datasets containing naturally blurred images, their sharp counterparts and the ground truth for the blur values.

Figure 5.34: Comparison of deblurring results using different blur maps using the images(13-22) provided in (D'ANDRES et al., 2016). Left column: original sharp image, blurry image. Remaining columns, from left to right: deblurring result using ground truth data and blur map of (ZHUO; SIM, 2011), (ZHANG; CHAM, 2012), (BAE; DURAND, 2007), (TANG; HOU; SONG, 2013), (SHEN; HWANG; PEI, 2012), (CHEN; CHEN; CHANG, 2016), (D'ANDRES et al., 2016) and the proposed method, respectively. (Best view can be seen zooming)

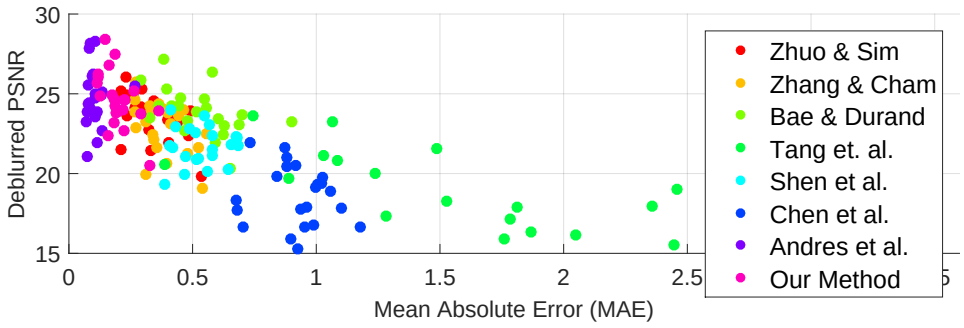


In this part of the dissertation, we perform a correlation analysis for the dataset proposed in (D'ANDRES et al., 2016) considering all methods used in the comparisons shown in the previous sections. In the first analysis, we compare the pairs MAE $\times$ PSNR considering all images and methods (22 images $\times$ 8 methods $=$ 174 pairs). More precisely, the Pearson correlation coefficient was -0.66, with confidence $p < 0.001$, indicating a statistically significant negative correlation between the MAE and PSNR, which was expected (better blur maps should yield better deblurred images).

We have also analyzed the relationship between the MAE and PSNR for each method individually. The scatter plots are shown in Fig. 5.35, and for most methods the PSNR tends to increases as the MAE decreases. In fact, the Pearson correlation coefficient was negative (around -0.5) and statistically significant ($p < 0.05$) for all methods except for (SHEN; HWANG; PEI, 2012), (CHEN; CHEN; CHANG, 2016) and (D'ANDRES et al., 2016). For these last three methods, the p-values were 0.84, 0.46 and 0.82, which highly suggests that there is no correlation between MAE and PSNR for these methods. We have also analyzed Spearman's and Kendall's rank correlation coefficients, which measures how well the variables are related by a monotonic function, and the results were consistent with the Pearson coefficient. Although we do not have a definitive explanation for this fact, we believe that the spatial distribution of the error across different methods might play an important role. For example, blur estimation errors along homogeneous regions can have a small effect on the deblurred image, whereas the opposite happens along edges or textured regions.

It is also important to emphasize that these results are highly affected by the deblurring method (as well as the parameters used in the chosen method). However, it indicates that for some methods the MAE of the blur map may not be a good indicative of the quality of the corresponding deblurred image.

Figure 5.35: Scatter plot of MAE and PSNR for different methods



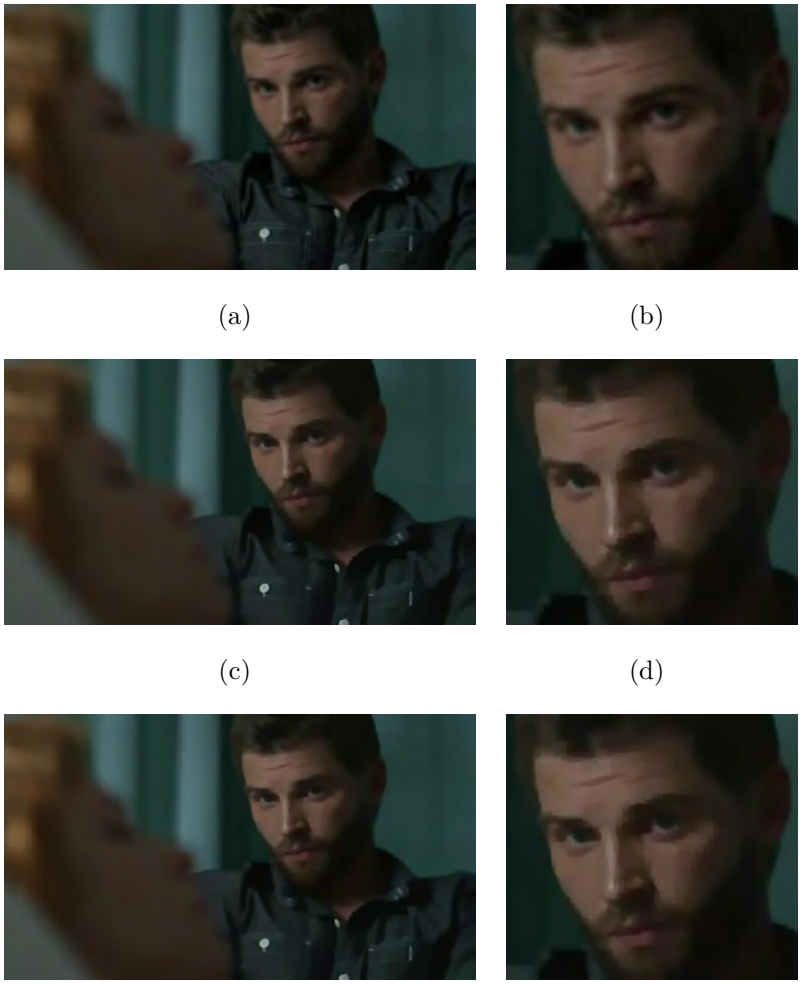
5.6.2 Video Deblurring

The last set of experiments aim to investigate the application of the proposed defocus blur estimation method for video sequences in the context of video deblurring. In the analysis, we used our defocus blur map estimation results with and without temporal coherence, and results are explored only quantitatively since (to our knowledge) there are no publicly available pairs of defocus/all-in-focus video sequences. In fact, we used the same video sequences evaluated in Section 5.4.

Some selected frames from the videos used in the experiments, along with deblurring results using our method with and without temporal coherence, can be seen in Figs. 5.36-5.39. The first observation is that for videos #1, #2, and #3, which were extracted from a TV Series, present visual artifacts (e.g. "blockiness" in Fig. 5.37(d)(f)), which can be caused by video compression. On the other hand, deblurring results of the video that we recorded (in Fig. 5.39) do not present these kinds of artifacts.

In terms of temporal artifacts, it is very difficult to evaluate the gain achieved by the temporal coherence based on still frames only. To see all deblurred results, please visit <https://sites.google.com/site/axkaraali/dissertationvideos>. However, visual evaluation shows that there is not much difference between the two deblurred version of videos, which indicates that the gain obtained by the temporal smoothness in the estimation does not seem to produce deblurred videos with less temporal artifacts.

Figure 5.36: Video deblurring results for Video #1. Original image (a) and the selected patch (b), deblurring results using our defocus estimation method with (c) and without temporal coherence (e) and corresponding selected patches (d)-(f).



(a) (b)

(c) (d)

(e) (f)

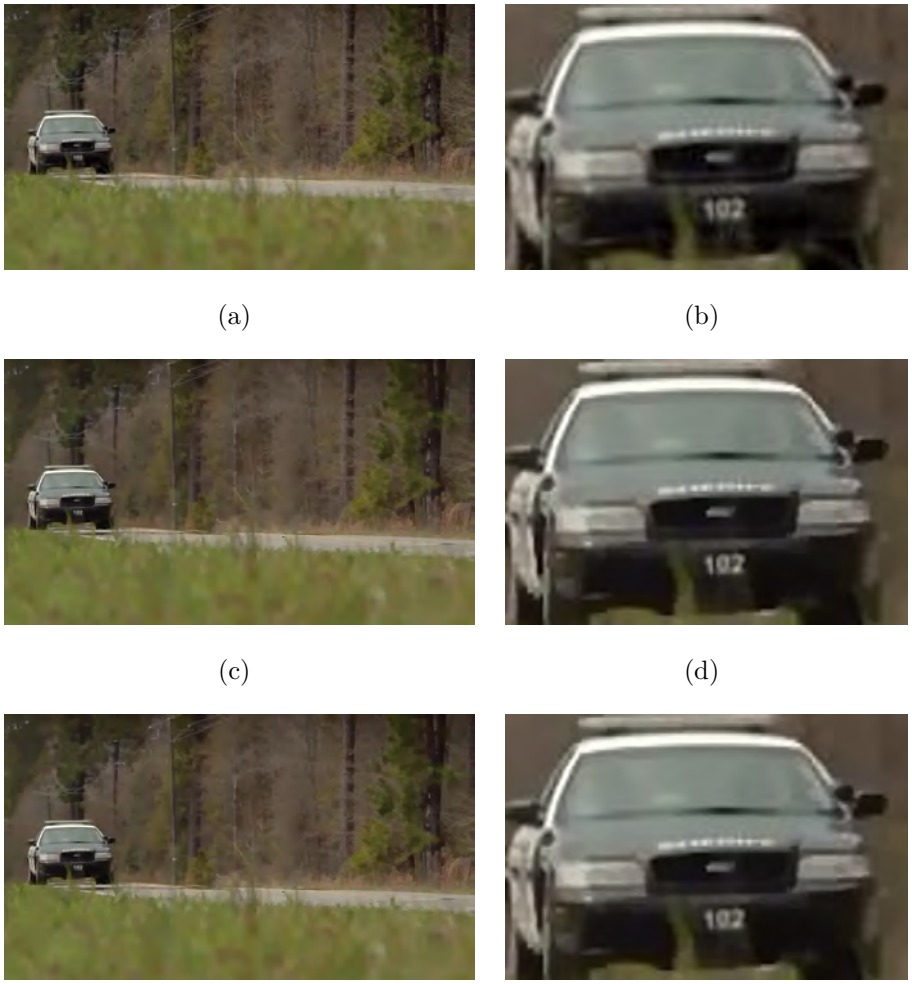
Figure 5.37: Video deblurring results for Video #2. Original image (a) and the selected patch (b), deblurring results using our defocus estimation method with (c) and without temporal coherence (e) and corresponding selected patches (d)-(f).

(a) (b)

(c) (d)

(e) (f)

Figure 5.38: Video deblurring results for Video #3. Original image (a) and the selected patch (b), deblurring results using our defocus estimation method with (c) and without temporal coherence (e) and corresponding selected patches (d)-(f).

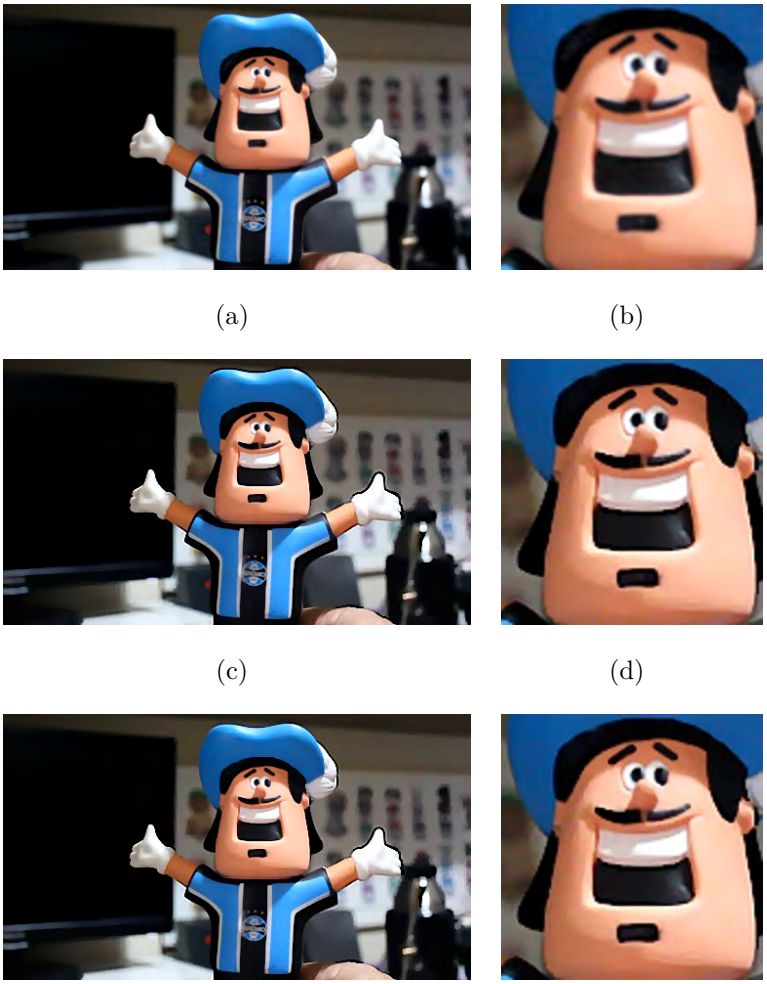


(a) (b)

(c) (d)

(e) (f)

Figure 5.39: Video deblurring results for Video #4. Original image (a) and the selected patch (b), deblurring results using our defocus estimation method with (c) and without temporal coherence (e) and corresponding selected patches (d)-(f).



(a) (b)

(c) (d)

(e) (f)

6 CONCLUSIONS

In this dissertation, we proposed two different edge-based defocus blur estimation methods for still images. The first approach computes 1D profiles along edge points orthogonal to the local contour, and evaluate the location of the edge (maximum of the derivative) to adaptively select the number of reblurring scales. Considering the time consumption of exploring 1D oriented edge profiles, a second method was proposed based on 2D multiscale image gradients, and local reblurring parameters were selected based on the agreement of an edge detector computed at several scales.

Since the mathematical formulation that supports both methods is based on continuous domain filters and the implementation uses discretized Gaussians, a correction step is proposed in order to deal with discretization errors. Experimental results showed that the refinement scheme was able to reduce blur estimation error in low-blur regions of synthetically blurred images, where error induced by the discretized Gaussian is larger.

The refined blur estimates are then smoothed locally along the image contours by minimizing an energy function that contains a data fidelity term and a regularization term, aiming to keep good initial blur estimates and correcting bad estimates. Our results showed that the proposed Connected Edge Filter (CEF) indeed improves the initial estimates both in terms of visual inspection and quantitative error analysis using the Mean Absolute Error (MAE).

The final full blur estimation methods were then compared with several other state-of-the-art defocus blur estimation methods using synthetic images, as in most blur estimation papers, and also using a database with naturally blurred images recently introduced by (D'ANDRES et al., 2016), which also contains "ground truth" data. Our experimental results showed that the proposed method presents a good compromise between running time and accuracy when compared to competitive approaches, being more accurate than all tested edge-based methods (ZHUO; SIM, 2011), (ZHANG; CHAM, 2012), (BAE; DURAND, 2007), (TANG; HOU; SONG, 2013), (SHEN; HWANG; PEI, 2012) and (CHEN; CHEN; CHANG, 2016) for the tests with synthetic images. Results using the recent natural database were consistent with the synthetic database: the proposed method was also more accurate than all edge-based methods, being inferior only to the patch-based approach

(D'ANDRES et al., 2016), which is based on Regression Tree Fields that tend to be computationally expensive.

This dissertation also presented an extension of the proposed methods based on still images to deal with defocused video sequences aiming to impose temporal consistency and reduce jittering, since applying defocus blur map estimation methods (including our) in a frame by frame manner tends to produce temporal inconsistencies. The temporal part of the proposed method is based on Kalman Filters, which inherently allow a compromise between data fidelity and temporal smoothing by adjusting the noise amounts in the state transition and observation models. In this work, the state transition model was set empirically in a way such that when the local appearance of a given pixel is consistent in time, the noise is small (so that smoothing is increased). On the other hand, when the observed defocus blur estimate (i.e. the proposed method for still images applied in a framewise manner) at a given pixel is considered as reliable, the observation noise is set to a smaller value aiming to maintain data fidelity. The final result is then a compromise between data fidelity and smoothness, a temporal analogous to the spatial compromise provided by the CEF. Experimental results show that the proposed model based on Kalman filtering provided smooth temporal trajectories, at the same time allowing sharp transitions during occlusions/disocclusions. Since the proposed scheme is applied in a pixel-wise manner, results tend to be better for static (or slowly moving) cameras. For video sequences with strong camera motion, the local patch analysis across frames tends produce large values, which reduces temporal filtering. In those cases, a more adequate solution would be to perform filtering along pixel paths, obtained by optical flow or particle tracking. However, our attempts for particle tracking in blurry videos were not successful, probably due to the weak textural information at blurry regions.

As an additional contribution, this dissertation also proposed a new image retargeting method for photos taken by shallow DoF cameras. To accomplish image retargeting, the estimated blur map is used to identify in-focus regions first, and cropping is applied to remove image boundaries. If cropping is not enough to meet the desired dimensions of the output image, improved seam carving is applied using a novel blur-aware energy function. Our experimental results showed that in-focus regions are indeed better preserved by our approach when compared to other state-of-the art methods e.g. (WANG et al., 2008), Multi-operator (RUBINSTEIN;

SHAMIR; AVIDAN, 2009), Improved Seam Carving (RUBINSTEIN; SHAMIR; AVIDAN, 2008), Nonhomogeneous warping (WOLF; GUTTMANN; COHEN-OR, 2007).

Finally, we have also explored the proposed blur map estimation method in the context of image and video deblurring. For still images, we used the natural dataset of (D'ANDRES et al., 2016), which contains blurry and all-in-focus images of the same scene, allowing an objective comparison. Using the same (fixed) non-blind deconvolution algorithm, we used the blur maps produced by several defocus blur estimation methods to estimate the sharp image. Our results showed that only the proposed method and (D'ANDRES et al., 2016) produced deblurred images with higher PSNR than the blurry image itself. For the same dataset, we have also performed a correlation analysis between the blur estimation error (MAE) and the quality of the deblurred image (PSNR). Our findings indicated that considering all images and methods, there is negative correlation with statistical significance, which was expected: the lower the MAE, the higher should be the PSNR. However, when analyzing each method individually, some of them (such as (D'ANDRES et al., 2016)) did not show significant correlation between MAE and PSNR. For video sequences, due to the lack of ground truth data, deblurring results are evaluated only visually. It was observed that regions presenting slight defocus blur were deblurred with minor visual artifacts, but the impact of temporal coherence could not be observed in these specific examples. Although qualitative evaluations showed that deblurring results with and without temporal coherence were very similar, the usage of different deblurring algorithms might produce better video deblurring results that could show the impact of temporal coherence. Also, temporal coherence could be included in the deblurring algorithm itself.

6.1 Future Work

As future work, firstly we intend to investigate a better way for estimating the initial sparse map by improving the confidence parameter ξ while considering the non-connected but neighboring edges for filtering the obtained sparse blur map. Clearly, a better estimate for ξ will also impose a better "guess" for the measurement noise parameter of the Kalman Filter. On the other hand, the current Kalman filtering model works in a pixel-wise manner, so that the temporal estimates at a

given pixel are smooth, by might introduce some spatial fluctuations. To better handle this issue, a vector version of the Kalman filter within image patches that also imposes spatial smoothness constraints in the temporal filtering might produce better results.

Another improvement that can be made is in the sparse blur map interpolation. Currently, nearly all the edge-based defocus blur estimation methods use the Laplacian based (KRISHNAN; FERGUS, 2009) interpolation scheme, which is slow and creates edges in the regions with uniform blur. To address this problem, we proposed to use of the fast guided filter (GASTAL; OLIVEIRA, 2011). There are several recent edge-aware filters (e.g. methods (HE; SUN; TANG, 2013; LI et al., 2016)) that could be explored in the context of joint-domain interpolation. In particular, since the sparse blur map is computed at image edges, it would be interesting to investigate the inclusion of gradient information in the guided filter.

Regarding image retargeting, we intend to use more recent seam carving algorithms in conjunction with our blur-aware energy function and the inclusion of temporal consistency information to deal with video retargeting applications exploring recent video carving methods such as (FURUTA; TSUBAKI; YAMASAKI, 2016).

Finally, as we stated before, there is no standard dataset to evaluate spatially varying defocus blur estimation methods. Therefore, most methods typically use artificially blurred images to perform quantitative evaluations. Recently, (D'ANDRES et al., 2016) provided a dataset containing 22 images with "known" blur values produced using the Lytro camera. Similarly, an extended version of this database with higher resolution and in different environments (e.g. indoor areas) could be produced in order to close the gap of lacking validation dataset.

APPENDIX A - RESUMO EXPANDIDO

Esta tese apresenta dois métodos diferentes para estimação de desfocagem usando uma única imagem. Ambos os métodos assumem uma função de espalhamento de ponto (*Point Spread Function* - PSF) Gaussiana e exploram a razão de magnitudes de gradientes de versões re-borradas da imagem original com escalas diferentes nas bordas da imagem, o que fornece uma expressão matemática fechada para borramento local.

Mais precisamente, considere um sinal 1D f contendo uma borda (descontínua) com amplitude a e deslocamento b em $x = 0$:

$$f(x) = au(x) + b, \tag{6.1}$$

onde u é a função degrau unitário.

Considerando um borramento uniforme, caracterizado por uma PSF Gaussiana espacialmente invariante $g(x; \sigma_b)$ com variância σ_b^2, a borda borrada f_b em um sinal de domínio contínuo é dada por

$$f_b(x) = f(x) * g(x; \sigma_b), \tag{6.2}$$

e então sua derivada é dada por

$$\begin{aligned} f_b'(x) &= (f(x) * g(x; \sigma_b))' = f(x) * g'(x; \sigma_b) \\ &= g(x; \sigma_b) = \frac{a}{\sqrt{2\pi\sigma_b^2}} \exp\left(-\frac{x^2}{2\sigma_b^2}\right) \end{aligned} \tag{6.3}$$

Se $f_{b_1}(x)$ é uma versão re-borrada de $f_b(x)$ com outro núcleo Gaussiano $g(x; \sigma_1)$ cujo o parâmetro de escala σ_1 é conhecido, então a razão R_g das derivadas no local da borda é dada por

$$R_g = \frac{|f_b'(0)|}{|f_{b_1}'(0)|} = \sqrt{\frac{\sigma_1^2 + \sigma_b^2}{\sigma_b^2}}, \tag{6.4}$$

de modo que o parâmetro de desfocagem σ_b da imagem observada pode ser obtido como

$$\sigma_b = \frac{1}{\sqrt{R_g^2 - 1}}\sigma_1. \tag{6.5}$$

De fato, essa abordagem foi a essência do método de estimação de borramento por desfoco proposta em (ZHUO; SIM, 2011).

A formulação apresentada até agora permite a estimação de borramento local por desfoco utilizando uma versão re-borrada do sinal observado $f_b(x)$. Uma extensão simples envolve o uso de duas versões re-borradas. Mais precisamente, se $f_{b_1}(x)$ e $f_{b_2}(x)$ são versões re-borradas de f_b usando núcelos Gaussianos com variâncias σ_1^2 e σ_2^2, respectivamente, a razão R_g das derivadas pode ser computada explicitamente como

$$R_g(x) = \frac{|f'_{b_1}(x)|}{|f'_{b_2}(x)|} = \sqrt{\frac{\sigma_b^2 + \sigma_2^2}{\sigma_b^2 + \sigma_1^2}} \exp\left(-\frac{(\sigma_2^2 - \sigma_1^2)x^2}{2(\sigma_b^2 + \sigma_2^2)(\sigma_b^2 + \sigma_1^2)}\right). \tag{6.6}$$

Se avaliarmos $R_g(x)$ no local exato da borda (i.e. em $x = 0$) e resolvemos Eq. 6.6 para σ_b, obtemos

$$\sigma_b = \sqrt{\frac{\sigma_2^2 - R_g(0)^2\sigma_1^2}{R_g(0)^2 - 1}}, \tag{6.7}$$

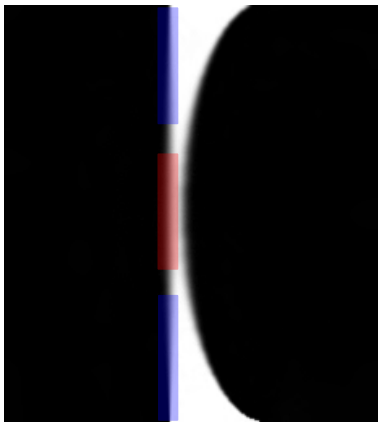
assumindo que $\sigma_2 > \sigma_1$.

A formulação usando uma ou mais versões re-borradas de f_b foi explorada por vários métodos, tais como (ZHANG; CHAM, 2012; KRIENER; BINDER; WILLE, 2013; JIANG et al., 2013; ZHANG et al., 2016; MAHMOUDPOUR; KIM, 2016; CHEN; CHEN; CHANG, 2016). Embora a idéia seja simples e rápida, é importante observar que nenhum desses artigos fornece diretrizes sobre como selecionar o(s) parâmetro(s) de re-borramento. É interessante notar que a Eq. 6.6 é afetada principalmente por três fatores ao lidar com imagens naturais: i) ruído, ii) exatidão da localização da borda, e iii) influência de bordas vizinhas, o que pode levar a escolhas conflitantes para o parâmetro de re-borramento. Neste trabalho, mostramos primeiro como σ_1 e σ_2 impactam a estimativa do borramento σ_b, e então apresentamos uma abordagem para selecionar esses parâmetros adaptativamente.

Para ilustrar o impacto das escalas de re-borramento, vamos considerar a imagem sintética artificialmente desfocada (mostrada na Fig. 6.1), cujo parâmetro de desfocagem σ_b varia verticalmente (na Eq. 6.2) a partir do topo ($\sigma_b = 0,5$) e aumenta linearmente até ($\sigma_b = 3$) no meio e volta a cair linearmente para ($\sigma_b = 0,5$). Na parte superior e inferior da imagem (realçadas em azul), as bordas estão bem separadas, mas no meio (realçado com vermelho) a estrutura é fina. Nós também produzimos versões ruidosas da Fig. 6.1 usando ruído gaussiano aditivo com médio zero, cujo desvio padrão é dado por $\eta = 1,275$ e $\eta = 2,55$.

Realizamos um conjunto de experimentos mantendo $\sigma_1 = 1$ fixo e focamos

Figure 6.1: Imagem sintética artificialmente desfocada.



o impacto do parâmetro de re-borramento σ_2. Em seguida, avaliamos média de erro absoluto (MEA) da estimatação de borramento por desfoco ao longo da parte superior e inferior da imagem (regiões azuis na Fig. 6.1), da parte central (região vermelha na Fig. 6.1) e ao longo de toda a borda esquerda (vertical), usando diferentes valores (globais) para σ_2. Os resultados, mostrados na Tabela 6.1, indicam que (como dissemos antes) escolher um valor adequado para σ_1 e σ_2 globalmente não é possível. Também pode ser observado que valores maiores para σ_2 produzem melhores resultados nas regiões azuis, que contêm pouca interferência de borda (e a diferença aumenta à medida que se adiciona mais ruído), enquanto que valores menores para σ_2 são melhores na região vermelha (onde a interferência de borda mais forte).

Table 6.1: Média de erros absolutos calculados em diferentes regiões de borda da imagem apresentada na Fig. 6.1 (Azul/Vermelho/Global) usando $\sigma_1 = 1$ e valores diferentes por σ_2 e variando a contaminação do ruído.

	Nível de ruído		
σ_2	$\eta = 0$	$\eta = 1,275$	$\eta = 2,55$
1.5	$0,065/\mathbf{0,224}/\mathbf{0,103}$	$0,065/\mathbf{0,266}/\mathbf{0,122}$	$0,081/\mathbf{0,270}/0,150$
2.0	$0,064/0,299/0,124$	$0,061/0,312/0,129$	$0,067/0,315/\mathbf{0,145}$
2.5	$\mathbf{0,063}/0,399/0,153$	$0,059/0,404/0,153$	$0,060/0,412/0,165$
3.0	$\mathbf{0,063}/0,521/0,191$	$\mathbf{0,057}/0,524/0,189$	$\mathbf{0,056}/0,533/0,199$

Nesta tese, foram propostos dois métodos diferentes para a estimatação de borramento por desfoco utilizando a escala adaptativa para re-borramento. O

primeiro explora os locais de borda estimados usando perfis de imagem 1D, enquanto o segundo usa gradientes de imagem diretamente.

- A primeira abordagem calcula perfis 1D ao longo de pontos de borda ortogonais ao contorno local, e avalia a localização da borda (máximo da primeira derivada) para selecionar adaptativamente o número de escalas no re-borramento.
- Um segundo método foi proposto com base em gradientes de imagem diretamente no domínio 2D, e os parâmetros de re-borramento locais foram selecionados com base na concordância de um detector de bordas calculado em várias escalas (considerando o consumo de tempo de explorar perfis de aresta orientados 1D).

Dada uma estimativa inicial da escala de desfocagem nas posições de borda proporcionada por qualquer um destes dois métodos, um novo método de filtragem local que suaviza as estimativas refinadas ao longo dos contornos de imagem também é proposto. Sendo σ_{b_i} a estimativa de borramento inicial em um pixel i, a filtragem é obtida pela minimização de

$$E = \sum_{i=1}^{N_c} \left[\xi_i (\alpha_i - \sigma_{b_i})^2 + \sum_{j=1}^{N_c} \psi_{ij} (\alpha_i - \alpha_j)^2 \right], \tag{6.8}$$

onde ξ_i é um valor de confiança para cada estimativa de borramento por desfoco σ_{b_i}, de modo que quando ξ_i é grande a estimativa refinada α_i tende a estar mais próxima da estimativa inicial σ_{b_i}. A filtragem é realizada em cada componente conexo do mapa de bordas da imagem, e N_c é o número de elementos no componente.

É também proposto um passo de correção que atenua os erros introduzidos pela discretização da formulação contínua e um filtro de domínio conjunto (*joint-domain filter*) rápido é explorado para propagar informações de desfocagem para toda a imagem, gerando o mapa de desfocagem completo.

Os resultados experimentais em imagens sintéticas e reais mostram que os métodos propostos apresentam resultados promissores para a estimativa de borramento por desfoco, com um bom compromisso entre qualidade e tempo de execução quando comparados a técnicas no estado-da-arte. Mais precisamente, isto pode ser visto na Fig. 6.2 que mostra imagens reais fornecidas em (D'ANDRES et al., 2016) para comparação visual e na a Tabela 6.2 apresenta uma avaliação quantitativa.

Para lidar com sequências de vídeo desfocadas, a consistência temporal também foi incluída no modelo proposto. Mais precisamente, Filtros de Kalman foram

Figure 6.2: Comparação de diferentes algoritmos de estimação de borramento utilizando imagens fornecidas em (D'ANDRES et al., 2016). Coluna esquerda: imagens originais. Colunas restantes, da esquerda para a direita: *ground truth* e resultados produzidos por (ZHUO; SIM, 2011), (ZHANG; CHAM, 2012), (BAE; DURAND, 2007), (TANG; HOU; SONG, 2013), (SHEN; HWANG; PEI, 2012), (CHEN; CHEN; CHANG, 2016), (D'ANDRES et al., 2016) e o método proposto, respectivamente.

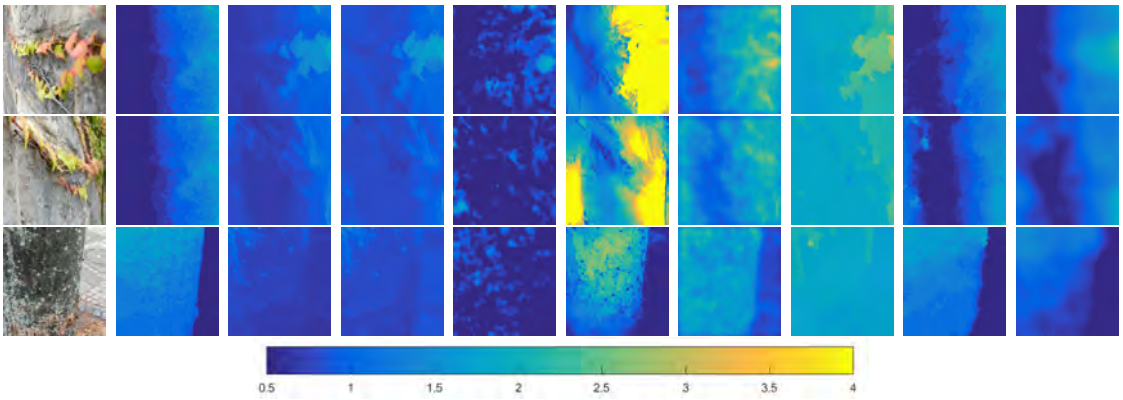


Table 6.2: Média de erros absolutos (MEA) para as imagens fornecidas em (D'ANDRES et al., 2016).

Imagem#	Zhuo&Sim	Zhang&Cham	Bae&Durand	Tang et.al.	Shen et.al.	Chen et.al.	Andres et.al.	*Nosso M.*
01	0,229	0,326	0,397	1,028	0,579	1,023	0,098	0,119
02	0,358	0,399	0,361	0,746	0,510	0,839	0,129	0,172
03	0,233	0,270	0,325	1,866	0,558	0,989	0,106	0,186
04	0,216	0,266	0,554	2,050	0,581	0,938	0,080	0,114
05	0,211	0,268	0,291	1,281	0,644	1,177	0,081	0,160
06	0,210	0,309	0,478	2,444	0,472	0,898	0,073	0,181
07	0,230	0,268	0,581	2,358	0,675	1,056	0,105	0,185
08	0,490	0,465	0,901	4,177	0,419	0,675	0,083	0,364
09	0,404	0,478	0,624	1,239	0,579	0,995	0,069	0,224
10	0,268	0,344	0,445	1,759	0,513	0,951	0,131	0,128
11	0,400	0,412	0,604	2,834	0,566	0,872	0,112	0,190
12	0,432	0,443	0,698	3,056	0,411	0,731	0,077	0,217
13	0,258	0,309	0,381	2,459	0,654	1,103	0,084	0,147
14	0,343	0,380	0,547	1,813	0,683	1,004	0,266	0,264
15	0,535	0,539	0,652	0,388	0,387	0,705	0,076	0,328
16	0,289	0,353	0,416	0,887	0,469	0,926	0,108	0,221
17	0,485	0,522	0,626	2,720	0,405	0,678	0,134	0,289
18	0,324	0,337	0,466	1,525	0,487	0,880	0,105	0,181
19	0,319	0,325	0,514	1,084	0,678	1,020	0,135	0,226
20	0,329	0,397	0,592	1,785	0,523	0,961	0,112	0,158
21	0,296	0,363	0,450	1,063	0,549	0,916	0,094	0,116
22	0,437	0,556	0,687	1,487	0,429	0,880	0,084	0,202
MEA Medio	0,332	0,379	0,527	1,821	0,535	0,919	0,106	0,199
Tempo Medio(Seg)	9,42	22,58	7,63	11,84	2,72	0,81	*n/a*	1,44

aplicados para gerar estimativas temporais suaves para cada pixel quando a aparência local da sequência de vídeo não varia muito, permitindo transições durante

mudanças drásticas da aparência local, que podem se relacionar com oclusões / desoclusões.

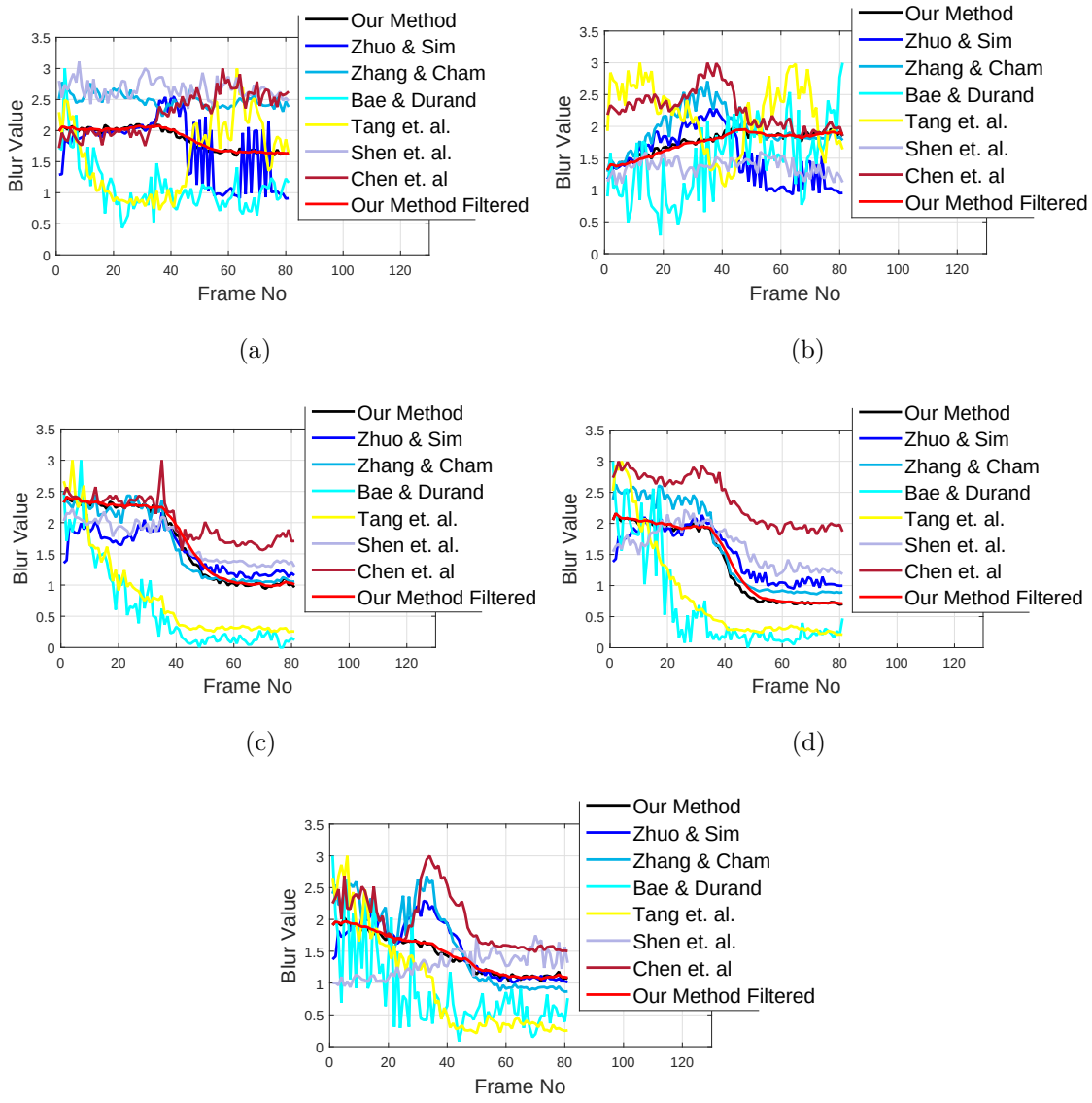
Uma experiência simples que inclui uma sequência de vídeo curta tomada por uma câmera estática e sem qualquer movimento visível do objeto, (somente as mudanças do plano focal) podem ser vistas na Fig. 6.3. Para mostrar os resultados estimados do mapa de desfocagem bem como a nossa versão filtrada de Kalman das sequências de mapa de desfocagem, escolhemos cinco pontos de controle (pontos vermelhos na Fig. 6.3) e avaliamos as escalas de desfocagem estimadas nesses pontos ao longo do tempo. Mais precisamente, os pontos #1 e #5 estão nas regiões mais desfocadas no início da sequência, e seu nível de borramento reduz ao longo do tempo. Os pontos #3 e #4 pertencem ao mesmo objeto e são inicialmente desfocados, ficando em foco após algum tempo, enquanto o ponto #2 apresenta o comportamento oposto, iniciando em foco e depois desfocando.

Figure 6.3: Sequências de vídeo desfocadas e cinco pontos de controle. (a) quadro #1, (b) quadro #32 e (c) quadro #80

(a) (b) (c)

A Fig. 6.4 mostra os valores estimados de borramento por desfoco para os cinco pontos de controle obtidos com diferentes métodos, bem como o método proposto com e sem coerência temporal. Uma vez que a variação espacial / temporal do vídeo borrado tem apenas mudanças focais ao longo do tempo, os mapas de borramento (completos) estimados deveriam mudar suavemente. No entanto, as estimativas de desfocagem na forma quadro a quadro (incluindo o nosso método) criam inconsistências temporais devido ao fato de que nenhum dos métodos usa informações temporais. Por outro lado, o esquema temporal proposto apresenta uma variação temporal suave no tempo (como na Fig. 6.4).

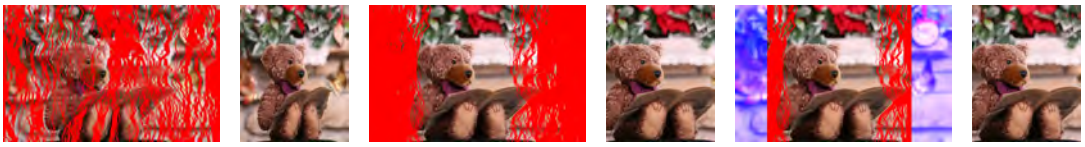
Figure 6.4: Valores estimados de borramento por desfoco para os cinco pontos de controle obtidos com diferentes métodos, bem como nosso método com e sem coerência temporal. (a) ponto de controle #1, (b) ponto de controle #2, (c) ponto de controle #3, (d) ponto de controle #4 and (e) ponto de controle #5



Finalmente, esta tese também mostra aplicações dos métodos propostos para a estimativa de desfocagem de imagem e vídeo. Um novo método de redimensionamento (*retargeting*) de imagens é proposto para fotografias capturadas por câmeras com baixa profundidade de campo. O método inclui informação de desfocamento local no contexto do método *seam carving*, visando preservar objetos em foco com melhor qualidade visual. Assumindo que os pixels em foco estejam relacionados às regiões de interesse de uma imagem com desfocamento, o método de redimensiona-

mento proposto começa com um método de corte (*cropping*), o qual remove as partes sem importância (borradas) da imagem, e então o método *seam carving* é aplicado com uma nova função de energia que prioriza as regiões em foco. Os resultados experimentais mostram que o método proposto funciona melhor na preservação de objetos em foco do que outras técnicas de redimensionamento de imagens.

Figure 6.5: (a)-(c) *Seams* usando (RUBINSTEIN; SHAMIR; AVIDAN, 2008) e nosso mapa de energia, respectivamente. (b)-(d) Imagens redimensionadas relacionadas com (a)-(c). (e) *Cropping* + *seams* usando nosso metodo completo, e (f) nosso resultado final.



(a) (b) (c) (d) (e) (f)

Para mostrar a uso do termo de borramento proposto, realizamos experimento comparando os *seams* produzidos pelo método utilizando apenas energia de gradiente (RUBINSTEIN; SHAMIR; AVIDAN, 2008) e nosso, que além de energia de gradiente, usa também a estimativa de borramento. O algoritmo subjacente é o mesmo, apenas a função de energia é alterada (para avaliar apenas os *seams* produzidos com e sem estimativa de borramento, o corte foi desativado neste exemplo). Os resultados são mostrados nas Fig. 6.5(a) e Fig. 6.5(c) respectivamente, enquanto as correspondentes imagens redimensionadas são mostradas nas Fig. 6.5(b) e Fig. 6.5(d), respectivamente. Como pode ser observado, os *seams* produzidos pelo nosso método evitam principalmente a distorção do objeto em foco, que está bem preservado na correspondente imagem redimensionada. O método baseado em gradiente, por outro lado, produz vários *seams* ao longo do *urso de pelúcia*, que gera uma imagem redimensionada deformada. Também ilustramos o resultado de nossa abordagem de redimensionamento completa, combinando corte e *seam* na Fig. 6.5(e), com a imagem redimensionada final mostrada em Fig. 6.5(f). Pode-se notar que apenas alguns *seams* adicionais devem ser removidos para alcançar a resolução desejada com a inclusão de corte, o que reduz o custo computacional (porque o corte é muito mais rápido do que a remoção dos *seams*).

Finalmente, esta tese também explora o método de estimação de desfocagem proposto no contexto de des-borramento de imagens e sequências de vídeo, e os

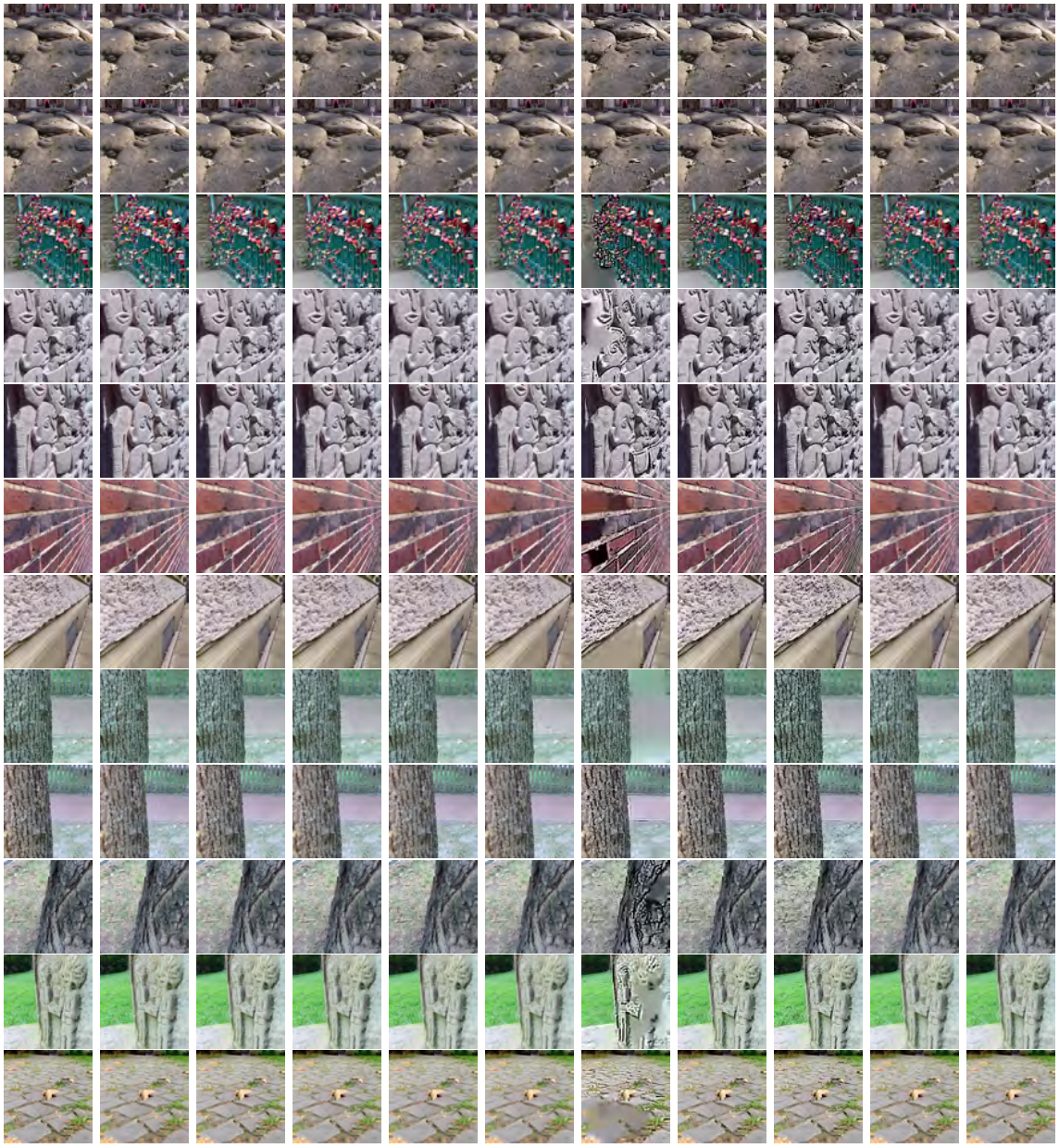
Table 6.3: Relação Sinal-Ruído de Pico das images desborradas fornecidas em (D'ANDRES et al., 2016)

Imagem#	Im. Borrada	Dado de GT	Zhuo&Sim	Zhang&Cham	Bae&Durand	Tang et.al.	Shen et.al.	Chen et.al.	Andres et.al.	*NossoM.*
01	25,51	26,48	24,90	23,77	25,31	21,14	22,35	19,75	26,25	26,07
02	24,54	25,19	24,18	23,79	24,22	23,62	22,53	19,80	24,90	24,89
03	23,74	25,23	23,60	22,88	23,51	16,35	20,12	16,75	24,76	24,21
04	23,58	25,98	24,87	24,61	24,10	16,17	21,12	17,74	25,56	25,64
05	26,50	28,32	25,17	23,85	25,82	17,35	20,28	16,65	27,83	26,81
06	24,05	25,06	21,47	19,95	23,30	15,52	21,04	15,92	23,87	24,66
07	27,09	28,59	26,02	25,70	26,37	17,98	22,32	18,87	28,28	27,49
08	23,47	24,48	23,95	23,97	23,23	19,82	21,61	18,30	24,40	23,92
09	22,56	23,35	21,94	21,23	22,41	20,00	21,49	19,13	23,26	22,70
10	23,97	25,13	24,09	22,14	24,25	15,92	20,89	16,63	24,93	24,87
11	23,47	24,04	23,37	23,11	23,43	17,12	23,04	21,63	23,87	23,81
12	23,73	24,49	23,70	23,62	23,65	19,24	23,97	21,95	24,36	23,96
13	28,09	28,41	24,95	23,25	27,17	19,02	21,80	17,80	28,13	28,40
14	24,75	25,17	24,52	24,05	24,70	17,92	21,77	19,32	25,47	25,19
15	20,43	21,17	19,84	19,04	20,32	20,57	19,33	16,66	21,08	20,48
16	23,87	25,14	24,16	21,64	24,26	19,70	19,95	15,28	25,01	24,60
17	24,33	24,10	22,40	21,60	23,01	18,84	21,76	17,69	22,69	23,75
18	22,71	23,73	22,74	22,40	22,70	18,24	22,82	20,99	23,53	23,17
19	24,31	25,47	24,11	24,22	23,84	20,81	22,23	19,38	25,11	24,61
20	22,23	22,63	21,46	20,63	21,91	17,13	20,94	17,86	21,92	22,38
21	25,63	26,47	25,26	24,36	24,76	23,23	23,61	20,52	26,08	26,23
22	23,91	24,24	23,24	22,50	23,04	21,54	22,90	20,41	24,06	23,95
PSNR Médio	24,20	25,13	23,63	22,83	23,88	18,96	21,72	18,59	24,79	24,63
Ganho Médio	*n/a*	0,92	−0,56	−1,37	−0,32	−5,23	−2,48	−5,60	0,58	0,42

resultados são comparados com vários outros métodos de estimação de desfocagem. As imagens des-borradas usando diferentes métodos de estimativa de borramento foram comparadas com a imagem original (sem borramento), e avaliadas quantitativamente usando a razão Sinal-Ruído de Pico (ou PSNR, de *Peak signal-to-noise ratio*). Os resultados mostrados na Tabela 6.3 indicam que apenas a técnica proposta e a recente abordagem de (D'ANDRES et al., 2016) alcançam ganhos no PSNR em comparação com a imagem original borrada.

A Fig. 6.6 mostra alguns exemplos de des-borramento utilizando os dados de *ground truth* e os mapas de desfocagem obtidos com diferentes métodos. Embora a análise visual seja muito subjetiva, pode-se observar que somente os resultados de des-borramentos pelo nosso mapa de desfocagem e pelo o mapa desfocagem de (D'ANDRES et al., 2016) estão próximos da imagem original, corroborando os resultados da Tabela 6.3.

Figure 6.6: Comparação dos resultados de des-borramento de imagens fornecidas em (D'ANDRES et al., 2016). Coluna esquerda: imagens originais e desfocadas. Colunas restantes, da esquerda para a direita: os resultados de des-borramento usando dado de *ground truth* e os mapas de desfocagem obtidos com os métodos de (ZHUO; SIM, 2011), (ZHANG; CHAM, 2012), (BAE; DURAND, 2007), (TANG; HOU; SONG, 2013), (SHEN; HWANG; PEI, 2012), (CHEN; CHEN; CHANG, 2016), (D'ANDRES et al., 2016) e do nosso método, respectivamente.



REFERENCES

AURICH, V.; WEULE, J. Non-linear gaussian filters performing edge preserving diffusion. In: **Mustererkennung 1995, 17. DAGM-Symposium**. London, UK, UK: Springer-Verlag, 1995. p. 538–545. ISBN 3-540-60293-3.

AVIDAN, S.; SHAMIR, A. Seam carving for content-aware image resizing. **ACM Trans. Graph.**, ACM, New York, NY, USA, v. 26, n. 3, jul. 2007. ISSN 0730-0301.

BAE, S.; DURAND, F. Defocus magnification. In: **Computer Graphics Forum**. [S.l.: s.n.], 2007. v. 26, p. 571–579.

BROX, T.; MALIK, J. Large displacement optical flow: Descriptor matching in variational motion estimation. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 33, n. 3, p. 500–513, March 2011. ISSN 0162-8828.

CANNY, J. A computational approach to edge detection. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, PAMI-8, n. 6, p. 679–698, Nov 1986. ISSN 0162-8828.

CHAKRABARTI, A.; ZICKLER, T.; FREEMAN, W. T. Analyzing spatially-varying blur. **2013 IEEE Conference on Computer Vision and Pattern Recognition**, IEEE Computer Society, Los Alamitos, CA, USA, v. 0, p. 2512–2519, 2010.

CHEN, D. J.; CHEN, H. T.; CHANG, L. W. Fast defocus map estimation. In: **2016 IEEE International Conference on Image Processing (ICIP)**. [S.l.: s.n.], 2016. p. 3962–3966.

CHIN, T. M.; KARL, W. C.; WILLSKY, A. S. Probabilistic and sequential computation of optical flow using temporal coherence. **IEEE Transactions on Image Processing**, v. 3, n. 6, p. 773–788, Nov 1994. ISSN 1057-7149.

D'ANDRES, L. et al. Non-parametric blur map regression for depth of field extension. **IEEE Transactions on Image Processing**, v. 25, n. 4, p. 1660–1673, April 2016.

ELDER, J. H.; ZUCKER, S. W. Local scale control for edge detection and blur estimation. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 20, p. 699–716, 1998.

FURUTA, R.; TSUBAKI, I.; YAMASAKI, T. Fast volume seam carving with multi-pass dynamic programming. In: **2016 IEEE International Conference on Image Processing (ICIP)**. [S.l.: s.n.], 2016. p. 1818–1822.

FüHR, G. et al. An evaluation of stereo matching methods for view interpolation. In: **IEEE International Conference on Image Processing**. [S.l.: s.n.], 2013. p. 403–407.

GASTAL, E. S.; OLIVEIRA, M. M. Domain transform for edge-aware image and video processing. In: ACM. **ACM Transactions on Graphics**. [S.l.], 2011. v. 30, n. 4, p. 69.

HE, K.; SUN, J.; TANG, X. Guided image filtering. In: _____. **Computer Vision – ECCV 2010: 11th European Conference on Computer Vision, Heraklion, Crete, Greece, September 5-11, 2010, Proceedings, Part I**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010. p. 1–14. ISBN 978-3-642-15549-9.

HE, K.; SUN, J.; TANG, X. Guided image filtering. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 35, n. 6, p. 1397–1409, June 2013. ISSN 0162-8828.

HOEFFKEN, M.; OBERHOFF, D.; KOLESNIK, M. Temporal prediction and spatial regularization in differential optical flow. In: _____. **Advanced Concepts for Intelligent Vision Systems: 13th International Conference, ACIVS 2011, Ghent, Belgium, August 22-25, 2011. Proceedings**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011. p. 576–585. ISBN 978-3-642-23687-7.

JANCSARY, J. et al. Regression tree fields - an efficient, non-parametric approach to image labeling problems. **2012 IEEE Conference on Computer Vision and Pattern Recognition**, IEEE Computer Society, v. 0, 2012.

JIANG, P. et al. Salient region detection by ufo: Uniqueness, focusness and objectness. In: **The IEEE International Conference on Computer Vision (ICCV)**. [S.l.: s.n.], 2013.

KARAALI, A.; JUNG, C. R. Adaptive scale selection for multiresolution defocus blur estimation. In: **Image Processing (ICIP), 2014 IEEE International Conference on**. [S.l.: s.n.], 2014. p. 4597–4601.

KARAALI, A.; JUNG, C. R. Image retargeting based on spatially varying defocus blur map. In: **2016 IEEE International Conference on Image Processing (ICIP)**. [S.l.: s.n.], 2016. p. 2693–2697.

KRIENER, F.; BINDER, T.; WILLE, M. Accelerating defocus blur magnification. In: . [S.l.: s.n.], 2013. v. 8667, p. 86671Q–86671Q–11.

KRISHNAN, D.; FERGUS, R. Fast image deconvolution using hyper-laplacian priors. In: **Proceedings of the 22Nd International Conference on Neural Information Processing Systems**. [S.l.: s.n.], 2009. (NIPS'09), p. 1033–1041. ISBN 978-1-61567-911-9.

LANG, M. et al. Practical temporal consistency for image-based graphics applications. **ACM Transactions on Graphics**, v. 31, n. 4, p. 34, 2012.

LANG, M. et al. Practical temporal consistency for image-based graphics applications. **ACM Trans. Graph.**, ACM, New York, NY, USA, v. 31, n. 4, p. 34:1–34:8, jul. 2012. ISSN 0730-0301.

LEVIN, A. et al. Image and depth from a conventional camera with a coded aperture. **ACM Trans. Graph.**, ACM, New York, NY, USA, v. 26, n. 3, jul. 2007. ISSN 0730-0301.

LEVIN, A.; LISCHINSKI, D.; WEISS, Y. Colorization using optimization. **ACM Transaction on Graphics**, ACM, New York, NY, USA, v. 23, n. 3, p. 689–694, aug. 2004.

LEVIN, A. et al. Understanding blind deconvolution algorithms. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 33, n. 12, p. 2354–2367, Dec 2011. ISSN 0162-8828.

LI, Y. et al. Deep joint image filtering. In: _____. **Computer Vision – ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV**. Cham: Springer International Publishing, 2016. p. 154–169. ISBN 978-3-319-46493-0.

LIU, F.; GLEICHER, M. Automatic image retargeting with fisheye-view warping. In: **Proceedings of the 18th Annual ACM Symposium on User Interface Software and Technology**. New York, NY, USA: ACM, 2005. (UIST '05), p. 153–162. ISBN 1-59593-271-2.

LIU, S.; ZHOU, F.; LIAO, Q. Defocus map estimation from a single image based on two-parameter defocus model. **IEEE Transactions on Image Processing**, v. 25, n. 12, p. 5943–5956, Dec 2016. ISSN 1057-7149.

LV, Z. et al. A continuous optimization approach for efficient and accurate scene flow. In: SPRINGER. **European Conference on Computer Vision**. [S.l.], 2016. p. 757–773.

MA, L. et al. Image retargeting quality assessment: A study of subjective scores and objective metrics. **IEEE Journal of Selected Topics in Signal Processing**, v. 6, n. 6, p. 626–639, Oct 2012. ISSN 1932-4553.

MAHMOUDPOUR, S.; KIM, M. Superpixel-based depth map estimation using defocus blur. In: **2016 IEEE International Conference on Image Processing (ICIP)**. [S.l.: s.n.], 2016. p. 2613–2617.

MATTHIES, L.; KANADE, T.; SZELISKI, R. Kalman filter-based algorithms for estimating depth from image sequences. **International Journal of Computer Vision**, Springer, v. 3, n. 3, p. 209–238, 1989.

PARIS, S.; KORNPROBST, P.; TUMBLIN, J. **Bilateral Filtering**. Hanover, MA, USA: Now Publishers Inc., 2009. ISBN 160198250X, 9781601982506.

PENTLAND, A. P. A new sense for depth of field. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, PAMI-9, n. 4, p. 523–531, July 1987.

PI, F. et al. Defocus blur estimation from multi-scale gradients. In: **Fifth International Conference on Machine Vision (ICMV 2012): Computer Vision, Image Analysis and Processing**. [S.l.: s.n.], 2012. v. 8783, p. 878309–878309–5.

REVAUD, J. et al. EpicFlow: Edge-Preserving Interpolation of Correspondences for Optical Flow. In: **Computer Vision and Pattern Recognition**. [S.l.: s.n.], 2015.

RUBINSTEIN, M. et al. A comparative study of image retargeting. **ACM Trans. Graph.**, ACM, New York, NY, USA, v. 29, n. 6, p. 160:1–160:10, dec. 2010. ISSN 0730-0301.

RUBINSTEIN, M.; SHAMIR, A.; AVIDAN, S. Improved seam carving for video retargeting. **ACM Trans. Graph.**, ACM, New York, NY, USA, v. 27, n. 3, p. 16:1–16:9, aug. 2008. ISSN 0730-0301.

RUBINSTEIN, M.; SHAMIR, A.; AVIDAN, S. Multi-operator media retargeting. **ACM Trans. Graph.**, ACM, New York, NY, USA, v. 28, n. 3, p. 23:1–23:11, jul. 2009. ISSN 0730-0301.

SAAD, E.; HIRAKAWA, K. Defocus blur-invariant scale-space feature extractions. **IEEE Transactions on Image Processing**, v. 25, n. 7, p. 3141–3156, July 2016. ISSN 1057-7149.

SCHARSTEIN, D.; SZELISKI, R. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. **International Journal of Computer Vision**, v. 47, n. 1, p. 7–42, 2002.

SHEN, C. T.; HWANG, W. L.; PEI, S. C. Spatially-varying out-of-focus image deblurring with l1-2 optimization and a guided blur map. In: **2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)**. [S.l.: s.n.], 2012. p. 1069–1072.

SHI, J.; XU, L.; JIA, J. Just noticeable defocus blur detection and estimation. In: **Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on**. [S.l.: s.n.], 2015. p. 657–665.

SPIELMAN, D. A.; TENG, S.-H. Nearly linear time algorithms for preconditioning and solving symmetric, diagonally dominant linear systems. **SIAM Journal on Matrix Analysis and Applications**, SIAM, v. 35, n. 3, p. 835–885, 2014.

SUH, B. et al. Automatic thumbnail cropping and its effectiveness. In: **Proceedings of the 16th Annual ACM Symposium on User Interface Software and Technology**. New York, NY, USA: ACM, 2003. (UIST '03), p. 95–104. ISBN 1-58113-636-6.

TAI, Y.-W.; BROWN, M. S. Single image defocus map estimation using local contrast prior. In: **2009 16th IEEE International Conference on Image Processing (ICIP)**. [S.l.: s.n.], 2009. p. 1797–1800. ISSN 1522-4880.

TANG, C.; HOU, C.; SONG, Z. Defocus map estimation from a single image via spectrum contrast. **Opt. Lett.**, OSA, v. 38, n. 10, p. 1706–1708, May 2013.

TOMASI, C.; MANDUCHI, R. Bilateral filtering for gray and color images. In: **Sixth International Conference on Computer Vision (IEEE Cat. No.98CH36271)**. [S.l.: s.n.], 1998. p. 839–846.

VAQUERO, D. et al. A survey of image retargeting techniques. In: . [S.l.: s.n.], 2010. v. 7798, p. 779814–779814–15.

VOLZ, S. et al. Modeling temporal coherence for optical flow. In: **2011 International Conference on Computer Vision**. [S.l.: s.n.], 2011. p. 1116–1123. ISSN 1550-5499.

VU, C. T.; PHAN, T. D.; CHANDLER, D. M. S3 : A spectral and spatial measure of local perceived sharpness in natural images. **IEEE Transactions on Image Processing**, v. 21, n. 3, p. 934–945, March 2012. ISSN 1057-7149.

WANG, Y.-S. et al. Optimized scale-and-stretch for image resizing. **ACM Trans. Graph.**, ACM, New York, NY, USA, v. 27, n. 5, p. 118:1–118:8, dec. 2008. ISSN 0730-0301.

WITKIN, A. Scale space filtering. In: **Proc. Eight International Joint Conference on Artificial Intelligence**. [S.l.: s.n.], 1983. p. 1019–1021.

WOLF, L.; GUTTMANN, M.; COHEN-OR, D. Non-homogeneous content-driven video-retargeting. In: **2007 IEEE 11th International Conference on Computer Vision**. [S.l.: s.n.], 2007. p. 1–6. ISSN 1550-5499.

YANG, Q. Stereo matching using tree filtering. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 27, n. 4, p. 834–846, 2014.

YI, X.; ERAMIAN, M. Lbp-based segmentation of defocus blur. **IEEE Transactions on Image Processing**, v. 25, n. 4, p. 1626–1638, April 2016. ISSN 1057-7149.

ZHANG, W.; CHAM, W. K. Single-image refocusing and defocusing. **IEEE Transactions on Image Processing**, v. 21, n. 2, p. 873–882, Feb 2012. ISSN 1057-7149.

ZHANG, X. et al. Spatially variant defocus blur map estimation and deblurring from a single image. **Journal of Visual Communication and Image Representation**, v. 35, p. 257 – 264, 2016. ISSN 1047-3203.

ZHU, X. et al. Estimating spatially varying defocus blur from a single image. **Image Processing, IEEE Transactions on**, v. 22, n. 12, p. 4879–4891, 2013. ISSN 1057-7149.

ZHUO, S.; SIM, T. Defocus map estimation from a single image. **Pattern Recognition**, v. 44, n. 9, p. 1852 – 1858, 2011.