Depth Estimation Using Segmentation in Natural Images

Shrayan Roy, Roll No: MD2220

Project Supervisor: Dr. Deepayan Sarkar*

Abstract

Estimating depth map from a single photograph is an intriguing problem with many interesting practical applications. Levin et al. [4] proposed a method which requires a modified camera with a specialized coded aperture. However, this approach is not applicable in practice since it requires modifying the camera before capturing the image, which may not always be feasible. We aim to explore whether similar tasks can be done using standard cameras. We have formulated simple parametric models for depth blurring, which we hope will lead to more efficient estimates.

Keywords: Auto Regressive Prior, Blur Kernel, Depth Map, Image Segmentation

1 Introduction

Traditional photographs are two dimensional projections of a three dimensional scene. The third dimension is *depth*, which represents the distance between camera lens and objects in the image. Depth perception is crucial for understanding spatial relationships, with various applications in computer vision tasks. Additionally, photography and cinematography benefit from depth perception, aiding in the creation of visually compelling compositions.

Most methods to estimate depth involve analyzing multiple images of the same scene to measure pixel sharpness across the stack and determine depth based on the distance from the sharpest pixel [2]. Hardware-based solutions are also available, usually involving extra apparatus such as light emitters.

Depth estimation based on single image is a more challenging problem because we have single observation for each pixel of the scene. Depth estimation from defocus blur exploits the phenomenon where objects appear more blurred depending on their distance from the camera lens, serving as a depth surrogate. Levin et al. [4] utilized this idea with a *sparse gradient prior* on natural images to estimate the *level of blur* per pixel. However, this method requires a modified

*Theoretical Statistics and Mathematics Unit, ISI Delhi < deepayan@isid.ac.in >

camera with a special coded aperture. Zhu et al. [8] employed Gabor filters for local frequency component analysis and utilized a *simple gradient prior*, without the need for a special coded aperture. After estimating depth for each pixel, an energy minimization technique based on Markov Random Field over the image is used to generate a smooth depth map.

In this project, we will explore whether a somewhat different approach can be used for the same problem. Instead of using a MRF approach for depth segmentation, we plan to start with modern high-performance segmentation algorithms such as Segment Anything [3]. Our hope is that if blurring is uniform within each segmented portion, it will be easier to estimate the blur kernel without a special coded aperture. We follow the approach developed by Nandy [5] to estimate the blur kernel, using in particular the locally dependent gradient prior proposed by them. However, instead of their nonparametric approach, we formulate and use simple parametric models for depth blurring, which we hope will lead to more efficient estimates for smaller image segments.

Figure 1: Example of Depth Map from [4]. Original image (Left panel) and corresponding Depth map (right panel)

2 Mathematical Formulation

When light rays spread from a single point source and hit the camera lens, they should ideally get refracted and converge on the pixel corresponding to the original scene. However, if the source is out of focus, the refracted rays spread out over neighboring pixels as well. This spreading pattern, determined by the object's distance from the lens or camera movement, is called the *Point Spread Function* (PSF) or *Blur Kernel*. The blurred image can be viewed as the result of convolving the original sharp image using the PSF. If we assume that the scene remains static for the duration of the photograph and there is no significant camera shake or rotation, then the observed blurred image $\boldsymbol{b}$ of dimension $M \times N$ can be modeled as:

$$\boldsymbol{b} = \boldsymbol{k} \otimes \boldsymbol{l} + \boldsymbol{\epsilon} \tag{1}$$

Where $\boldsymbol{k}$ is an $m \times n$ blur kernel, $\boldsymbol{l}$ is the $(M+m) \times (N+n)$ *true latent image* which we want to

estimate, $\boldsymbol{\epsilon}$ is an $M \times N$ matrix of noise, and $\otimes$ denotes the *convolution* operator (By *convolution* we mean *valid* convolution). Reconstructing the latent image from an observed blurred image is called the **image deconvolution problem**. Depending on whether the blur kernel is known or unknown, we classify it as *non-blind deconvolution* or *blind deconvolution* problem. In either case, it is an ill-posed problem because the number of parameters is larger than the number of observations MN. One solution to this is to assume a prior for the latent image $\boldsymbol{l}$.

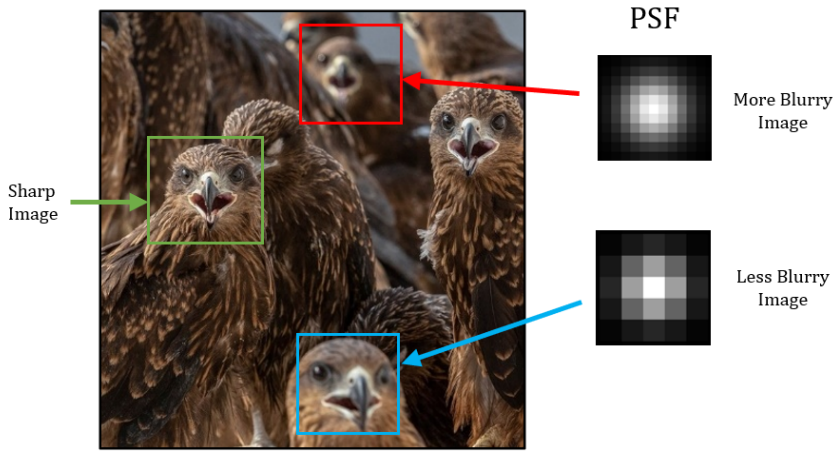

Figure 2: Example of Spatially Varying Blur: Each patch exhibits a distinct level of blur.

Note that, the model defined above assumes that the associated PSF is *shift invariant*, meaning the same PSF applies to all pixels of the underlying latent image. However, this may not be the case (Figure 2). In the context of depth from defocus blur, the PSF function is *not* shift invariant i.e. it is *spatially varying*. Therefore, (1) will not hold. We redefine it as:

$$\boldsymbol{b}[\boldsymbol{t}] = (\boldsymbol{k_t} \otimes \boldsymbol{l})[\boldsymbol{t}] + \boldsymbol{\epsilon}[\boldsymbol{t}] \tag{2}$$

Where $[t]$ indicates the corresponding elements at pixel location $\boldsymbol{t}$ and $\boldsymbol{k_t}$ is the spatially varying blur kernel at pixel location $\boldsymbol{t}$. Now, the problem of estimating blur kernel and latent image becomes more ill-posed because for each pixel, we need to estimate a blur kernel. However, if the blurring is only due to the objects being away from the plane of focus, we can assume special structures of the associated blur kernels. We model the blur kernel $\boldsymbol{k_t}$ as some probability density over a square grid. For each pixel location $\boldsymbol{t}$, we characterize the blur kernel by the parameter $\boldsymbol{\theta_t}$ that determines the *scale* or *spread* of the associated probability distribution (e.g. scale parameters in bivariate normal distribution). This parameter $\boldsymbol{\theta_t}$ encompasses information regarding the level of blur, hence providing insight into depth. Our objective is to estimate this parameter $\boldsymbol{\theta_t}$ based on the observed blurred image $\boldsymbol{b}$ for each pixel location $\boldsymbol{t}$.

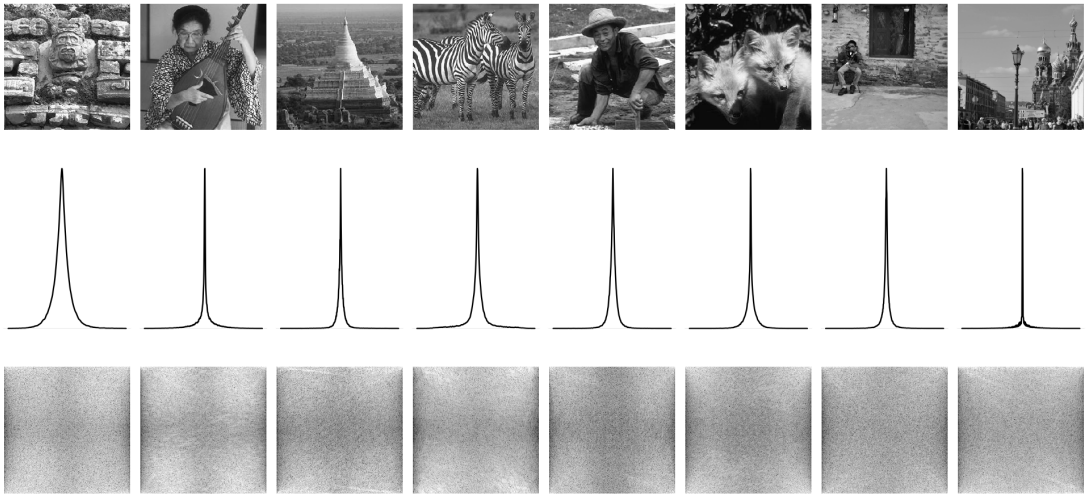

Figure 3: Eight sharp images (top row) and their density plot of horizontal gradients (middle row) and plot of log modulus DFT coefficients of horizontal gradients (bottom row). Similar observations are found for vertical gradients as well.

3 Priors on Natural Images

3.1 Choice of Prior and Basic Ideas

The prior family used for the latent *natural image*[1] is motivated by the observation that when a gradient filter is applied to image, the distribution of the output has a consistent and distinctive form across a wide range of scene types, with the distribution sharply peaked at zero and relatively heavier tails than the Gaussian distribution and Laplace distribution (Figure 3). Priors with these features are often referred to as *sparse priors* and a useful parametric family to model this is the so called **Hyper-Laplacian Distribution** given by

$$f_\alpha(z) = \frac{\alpha}{2\Gamma(\frac{1}{\alpha})}\exp(-|z|^\alpha), z \in \mathbb{R} \quad \text{and} \quad \alpha > 0 \tag{3}$$

For $\alpha = 2$ we have Gaussian distribution and for $\alpha = 1$ we have Laplace distribution. Levin et al. [4] used $\alpha = 0.8$, while Zhu et al. [8] utilized $\alpha = 2$, assuming IID gradients. Nandy [5] showed that the assumption of independent image gradients is incorrect, and modeled the dependency structure of gradients using a simple first-order AR Model.

To apply these priors, we must express the blur model in terms of image gradients, specifically in frequency domain. We will do this for (1). If $\boldsymbol{\delta_h} = [-1, 1]$ and $\boldsymbol{\delta_v} = [-1, 1]^T$, then the horizontal and vertical gradients are given by

$$\boldsymbol{\delta_h} \otimes \boldsymbol{b} = \boldsymbol{\delta_h} \otimes (\boldsymbol{k} \otimes \boldsymbol{l}) + (\boldsymbol{\delta_h} \otimes \boldsymbol{\epsilon}) = k \otimes (\boldsymbol{\delta_h} \otimes \boldsymbol{l}) + (\boldsymbol{\delta_h} \otimes \boldsymbol{\epsilon}) \tag{4}$$

[1] By *natural*, we refer to typical scenes captured in amateur digital photography, excluding specialized contexts like astronomy or satellite imaging.

$$\boldsymbol{\delta_v} \otimes \boldsymbol{b} = \boldsymbol{\delta_v} \otimes (\boldsymbol{k} \ \otimes \ \boldsymbol{l}) \ + \ (\boldsymbol{\delta_v} \otimes \boldsymbol{\epsilon}) = k \otimes (\boldsymbol{\delta_v} \ \otimes \ \boldsymbol{l}) \ + \ (\boldsymbol{\delta_v} \otimes \boldsymbol{\epsilon}) \tag{5}$$

To keep the notations simple, we will henceforth take the model by combining (4) and (5)

$$\boldsymbol{y} = \boldsymbol{k} \ \otimes \ \boldsymbol{x} \ + \ \boldsymbol{n} \tag{6}$$

Where, $\boldsymbol{y}$ is the horizontal (or, vertical) gradient of observed blurred image. $\boldsymbol{x}$ and $\boldsymbol{n}$ is the same for latent image and noise. By virtue of the *Convolution Theorem*, we rewrite (6) in the frequency domain as

$$\boldsymbol{Y} = \boldsymbol{K} \odot \boldsymbol{X} + \boldsymbol{N} \tag{7}$$

Where, $\boldsymbol{Y}, \boldsymbol{K}, \boldsymbol{X}$ and $\boldsymbol{N}$ are the *Discrete Fourier Transform*'s of $\boldsymbol{y}, \boldsymbol{k}, \boldsymbol{x}$ and $\boldsymbol{n}$ respectively. $\odot$ indicates the *element wise product* operator. $\forall\ \boldsymbol{\omega} = (\omega_1, \omega_2)$ we have

$$\boldsymbol{Y_\omega} = \boldsymbol{K_\omega} \boldsymbol{X_\omega} + \boldsymbol{N_\omega} \tag{8}$$

Remark: In practice the size of the blur kernel $\boldsymbol{k}$ is much smaller compared to the latent image $\boldsymbol{l}$ and $\boldsymbol{x}$. But for (7), the size of $\boldsymbol{K}$ must be the same as the size of $\boldsymbol{X}$. Thus, we pad $\boldsymbol{k}$ symmetrically with zeros to make of same size as $\boldsymbol{x}$ and then take DFT.

3.2 Prior on Fourier coefficients

If the elements of $\boldsymbol{x}$ are zero mean IID random variables with pdf (3). Then by orthogonality of DFT, $\boldsymbol{X_\omega}$'s must be IID complex normal when $\alpha = 2$ and uncorrelated and asymptotically complex normal when $\alpha \neq 2$ by CLT of Peligrad and Wu [6]. When elements of $\boldsymbol{x}$ are correlated, $\boldsymbol{X_\omega}$'s are still asymptotically independent and complex normal; however, depending on the correlation structure, the variance is no longer constant and depends on the specific frequency ω. We will use either the IID prior, or the simple AR prior of Nandy [5], for which $Var(\boldsymbol{X_\omega}) = g_\omega \sigma^2$ with the form of g_ω known explicitly.

For spatially varying blur, it is not immediately clear how we can express (2) in terms of image gradients. We need to assume that the blur kernel $\boldsymbol{k_t}$ is shift invariant in a neighborhood $\boldsymbol{\eta_t}$ of size $p_1(\boldsymbol{t}) \times p_2(\boldsymbol{t})$ containing $\boldsymbol{t}$. Then we get equation (9) as model for blur. This assumption is more or less true, because we expect objects in small local patches to have same depth and hence same level of blur (Figure 2). We apply the priors discussed above to these specific patches of the image.

$$\boldsymbol{y}[\boldsymbol{t'}] = (\boldsymbol{k_t} \ \otimes \ \boldsymbol{x})[\boldsymbol{t'}] \ + \ \boldsymbol{n}[\boldsymbol{t'}] \quad \forall \boldsymbol{t'} \in \boldsymbol{\eta_t} \tag{9}$$

4 Parametric Models for Blur Kernel

From a single point source, light rays emit in different directions and fall on the lens of camera. The diffracted rays by the lens then form a circular shape on the camera sensor plane, which is called the *Circle of Confusion* or *Blur Circle* (Fig 4). The diameter of the blur circle (c_{diam}) and the depth of an object (d) in a given camera setting are related as given in [1]:

$$c_{diam} = a_{diam} f \left| \frac{d - d_{focus}}{d(d_{focus} - f)} \right| \approx a_{diam} f \left| \frac{1}{d_{focus}} - \frac{1}{d} \right| \tag{10}$$

In a given camera setting (i.e., fixed a_{diam} and f)[2], $c_{diam} \propto \left| \frac{1}{d_{focus}} - \frac{1}{d} \right|$. As we move away from the plane of focus on either side, we encounter a similar type of c_{diam}. Thus, from the diameter of the blur circle, it is challenging to accurately estimate the depth d of an object in an image. Because for each value of c_{diam}, two possible values of d exist. It is evident from (10) that, the support of PSF in our case must be circular rather than square. Due to *diffraction* caused by the camera lens and the boundaries of the circular aperture, we expect the intensity distribution of light to be spherical symmetrically distributed over the circular support. Keeping all these in mind we propose the following models for blur kernel.

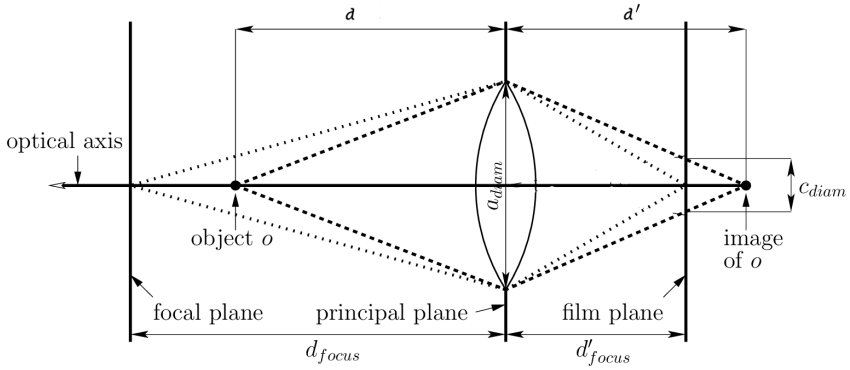

Figure 4: Thin Lens Model of Camera and Circle of Confusion c_{diam} in a given camera setting

1. Disc Kernel: It is the simplest model for blur kernel. The assumption being that uniform spread of light over disc. We characterize the kernel using the parameter r.

$$k(x, y) = \frac{1}{\pi r^2} \times \mathrm{I}_{\{x^2+y^2 \leq r^2\}}$$

2. Circular Gaussian Kernel: An obvious choice of kernel in any scientific study is the the widely used Gaussian/Normal distribution. Here, we will consider a truncated version of it over circular region. We characterize the kernel using the radius of circle r and scale parameter h.

$$k(x, y) = \frac{C_{h,r}}{2\pi h^2} e^{-\frac{x^2+y^2}{2h^2}} \times \mathrm{I}_{\{x^2+y^2 \leq r^2\}}$$

[2]For most cameras, $d_{focus} >> f$, hence the approximation.

3. Circular Cauchy Kernel: We know that the intercept on the x-axis of a beam of light coming from the point $(0, h)$ under certain assumption is distributed as a Cauchy$(0, h)$ distribution. Extending this concept to the two-dimensional case, we have a bivariate Cauchy distribution over a circular support. We characterize this using the radius of the circle r and the scale parameter h.

$$k(x, y) = \frac{C_{h,r}}{2\pi} \frac{h}{(x^2 + y^2 + h^2)^{3/2}} \times \mathrm{I}_{\{x^2+y^2 \leq r^2\}}$$

4. Rectangular Gaussian Kernel: We can also consider a truncated Gaussian kernel defined over a finite square grid S_h, characterized by h. In this scenario, we have only one parameter h.

$$k(x, y) = \frac{C_h}{2\pi h^2} e^{-\frac{x^2+y^2}{2h^2}} \mathrm{I}_{\{(x,y)\in S_h\}}$$

Remark: The parameter r in both the Circular Gaussian and Circular Cauchy kernel characterizes the radius of the blur circle (i.e., $c_{diam}/2$). For a given camera setting, there exists a relation between h and r, namely $h = \kappa \times r$ with κ depending upon particular camera. This implies that we cannot change h and r independently.

5 Maximum Likelihood Estimation of Blur Kernel Parameters

Our main focus in this section is to develop *Maximum Likelihood Estimation* procedure for the parameter $\boldsymbol{\theta_t}$ of blur kernel for each pixel location $\boldsymbol{t}$. For the first three kernels $\boldsymbol{\theta_t} = (r_t, h_t)$, and for the Gaussian kernel $\boldsymbol{\theta_t} = h_t$. We need to estimate these parameters based on the observed image $\boldsymbol{b}$ or equivalently $\boldsymbol{y}$. For that we need joint distribution of elements of $\boldsymbol{y}$.

As our priors for $\boldsymbol{x}$ are defined in terms of Fourier coefficients, we move to the frequency domain. In the case of generalized prior we have $\boldsymbol{X_\omega} \sim \mathcal{CN}(0, \sigma^2 g_\omega) \ \ \forall \boldsymbol{\omega}$ independently. In addition, if the errors in the original image, given by $\boldsymbol{\epsilon}$, are assumed to be IID Gaussian, then its gradient $\boldsymbol{n}$ will have correlated elements and successive elements in the direction of the gradient will have correlation 0.5, while all other pairs will be uncorrelated. This will induce a non-constant variance for $\boldsymbol{N_\omega}$, given by a function h_ω such that $Var(\boldsymbol{N_\omega}) = \eta^2 h_\omega$. Hence, $\boldsymbol{N_\omega} \sim \mathcal{CN}(0, \eta^2 h_\omega)$ independently. An explicit expression for h_ω can be found similarly to that of g_ω.

From (8) we have, $\boldsymbol{Y_\omega} \sim \mathcal{CN}(0, \sigma^2 |K_\omega|^2 g_\omega + \eta^2 h_\omega) \ \ \forall \boldsymbol{\omega}$. For a given choice of model for blur kernel, we estimate parameters $\boldsymbol{\theta}$ based on $\boldsymbol{Y_\omega}$'s using maximum likelihood principle. For ease of calculation we have used likelihood of $|\boldsymbol{Y_\omega}|^2$'s.

Result: *If $Z \sim \mathcal{CN}(0, \sigma^2)$. We know that $Re(Z)$ and $Im(Z)$ follows $\mathcal{N}(0, \frac{\sigma^2}{2})$ independently. Then, $|Z|^2 = Re^2(Z) + Im^2(Z) \sim \frac{\sigma^2}{2}\chi^2_2 \equiv Exp(\lambda = \frac{1}{\sigma^2})$.*

Using last result, $\boldsymbol{|Y_\omega|^2} \sim \mathrm{Exp}(\lambda_\omega = \frac{1}{\sigma^2 |K_\omega|^2 g_\omega + \eta^2 h_\omega}) \ \forall \boldsymbol{\omega}$ independently. If $f_{\theta,\omega}(.)$ denotes the pdf of $\mathrm{Exp}(\lambda_\omega = \frac{1}{\sigma^2 |K_\omega|^2 g_\omega + \eta^2 h_\omega})$ for given parameters $\boldsymbol{\theta}$ (say,) of blur kernel model. Then, likelihood of $\boldsymbol{|Y_\omega|^2}$ is given by -

$$f_\theta(|Y_\omega|^2, \forall\omega) = \prod_\omega f_{\theta,\omega}(|Y_\omega|^2) \tag{11}$$

The above considerations are only for gradients in a particular direction, i.e., *horizontal* or *vertical*. To incorporate both directions in the estimation procedure, we assume for simplicity that they are independent. The joint likelihood is then given by

$$L(\boldsymbol{\theta}) = L_h(\boldsymbol{\theta}) \times L_v(\boldsymbol{\theta}) = f_\theta(|Y_{h,\omega}|^2, \forall\omega) \times f_\theta(|Y_{v,\omega'}|^2, \forall\omega') \tag{12}$$

Our goal is to find the $\boldsymbol{\theta}$ maximizing $L(\boldsymbol{\theta})$, or equivalently $\log L(\boldsymbol{\theta}) = \log L_h(\boldsymbol{\theta}) + \log L_v(\boldsymbol{\theta})$. We do this by numerically optimizing the objective (log-likelihood) function. For the spatially varying case, we can simply apply this procedure to local patches $\boldsymbol{\eta_t}$ for all pixel locations $\boldsymbol{t}$ in the image domain, or to segments identified by segmentation algorithms.

6 Challenges in ML Estimation

The log-likelihood, defined in (12), is a complicated function of the blur kernel parameter $\boldsymbol{\theta}$. This parameter is involved in the expression λ_ω through $|K_\omega|^2$, which itself is a complicated function of $\boldsymbol{\theta}$. Therefore, before we start using any optimization method to find the maximizer of the log-likelihood, we should investigate the behavior of $L(\boldsymbol{\theta})$ as a function of $\boldsymbol{\theta}$. For this, we conducted simulated experiments as shown in Figure 5. For now, we will focus on the disc kernel. We considered a sequence of values for r ranging from 1 to 4 with a step size of $\Delta r = 0.05$, and for σ ranging from 0.01 to 0.4 with a step size of $\Delta\sigma = 0.01$, while keeping $\eta = 0.001$ constant.

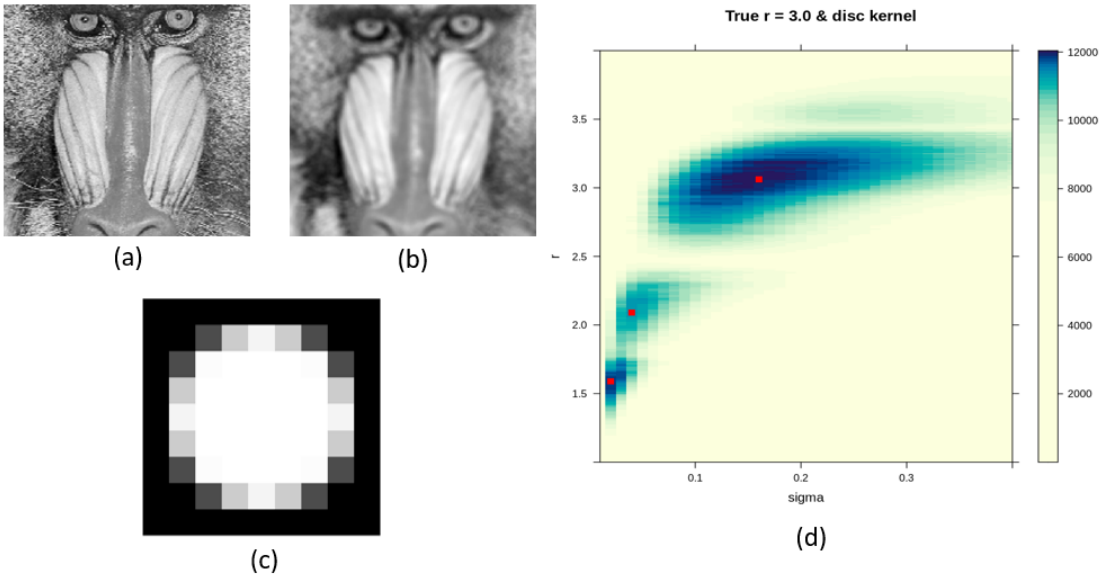

Figure 5: (a) 101 × 101 Sharp Image, (b) Blurred Image Using disc kernel with $r = 3$, (c) Disc Kernel with $r = 3$, (d) Levelplot of log likelihood as a function of σ and r

In Figure 5, (a) shows the original sharp image of size 101×101. We use a disc kernel with $r_{\text{true}} = 3$ to simulate defocus blur as shown in (b). The log likelihood is plotted as a function of r and σ in (d). Three local maxima points are visible in the plot, indicated by red dots. The global maximum is located near $r = 3$, with the corresponding value of σ around 0.17. Maximizing the log likelihood over a grid of values of (r, σ) gives a correct estimate of r in this case. However, it is not guaranteed that the global maximum always indicates the true parameter r. Figure 6 shows another example, where we implement the same simulation with $r_{\text{true}} = 2.5$. But this time the global maximum is attained at $r = 1.65$ with corresponding $\sigma = 0.04$, which is different from the true parameter value. In this case, we have three local maxima, with one near the true parameter value and corresponding σ near 0.2.

Simply looking for global maxima is not the solution. Choice of the prior parameter σ significantly affects the performance of ML Estimation. In the Bayesian Paradigm, this is quite common. If we can choose σ such that the maximizer of the log-likelihood for that given σ closely matches the true parameter, then we are done. To find a reasonable value of σ we use simulation. We consider five sharp images of size 255×255 and true parameter values $r_{true} = 1, 3, 5$ to simulate defocus blur. For each r_{true} value and fixed patch size, we randomly select five patches from each image and plot the estimated r as a function of σ for each random patch. We have considered patch sizes 51×51, 101×101, and 201×201.

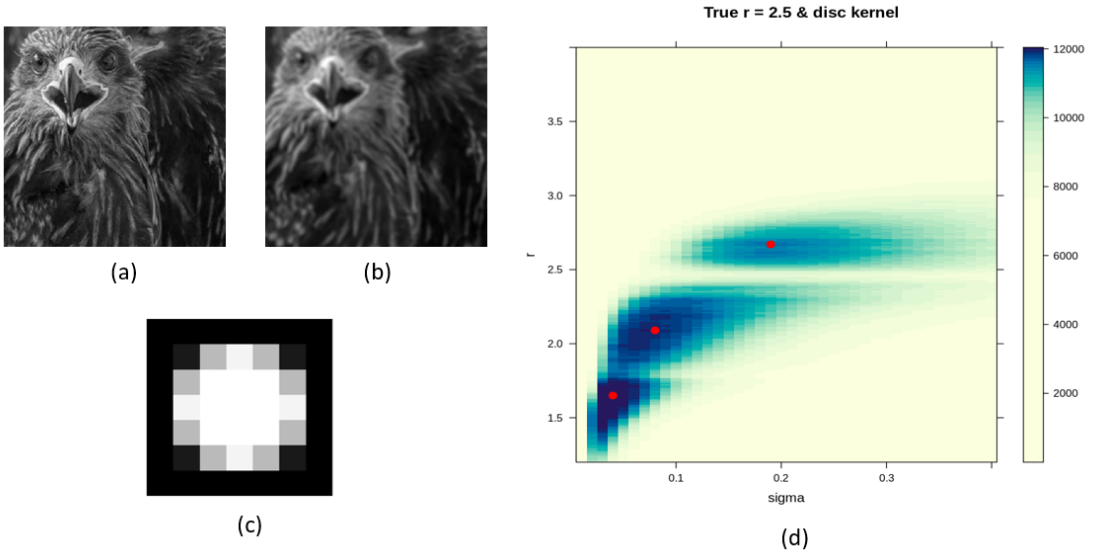

Figure 6: (a) 101×101 Sharp Image, (b) Blurred Image Using disc kernel with $r = 2.5$, (c) Disc Kernel with $r = 2.5$, (d) Levelplot of log likelihood as a function of σ and r

From Figure 7, 8 and 9, we observe that as patch size increases, the estimated r becomes more stable as a function of σ. Most importantly for $\sigma = 0.2$, the estimated r closely matches with the true value r_{true} across all cases, indicating that $\sigma = 0.2$ is a suitable choice for the prior parameter. This observation is consistent with Figures 5 and 6. It is also clear that the maximum likelihood estimation does not perform well for small patch sizes as expected. Because we have less number of pixels for small patch sizes.

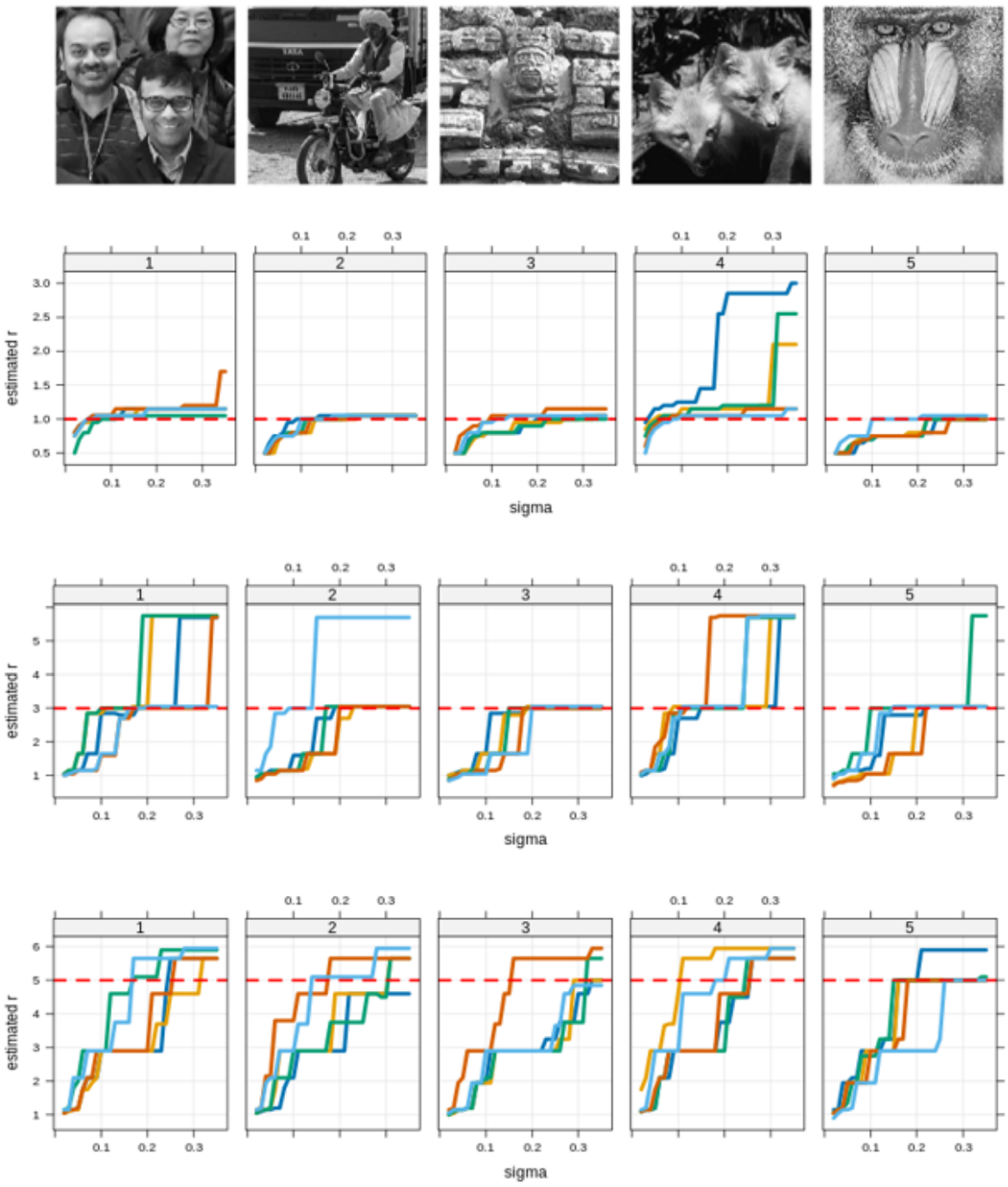

Figure 7: 51×51 Patch and Disc Kernel: Top row indicates the sharp images used for simulation and second, third and bottom row corresponds to r_{true} = 1,3,5 respectively.

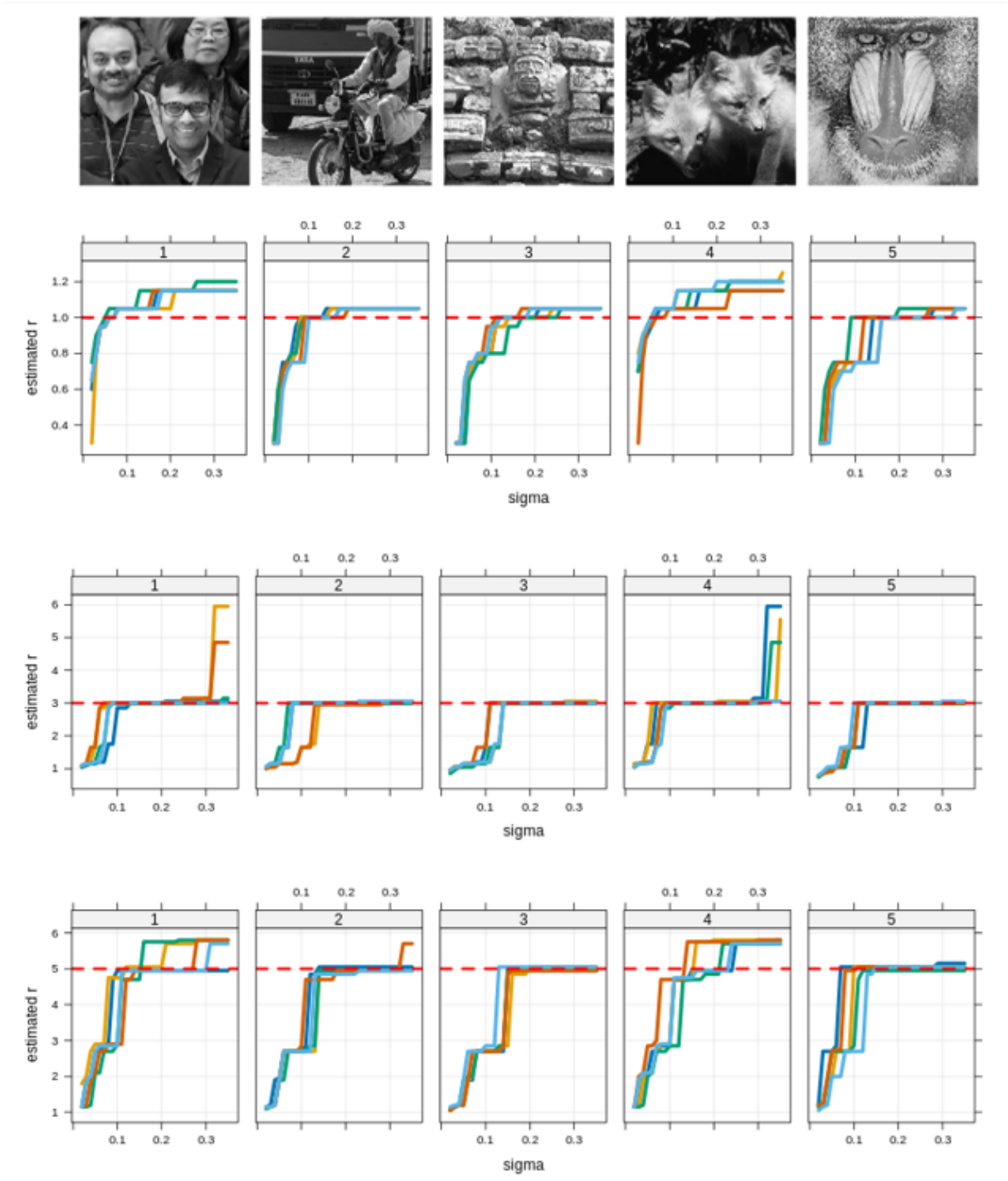

Figure 8: 101 × 101 Patch and Disc Kernel: Top row indicates the sharp images used for simulation and second, third and bottom row corresponds to r_{true} = 1,3,5 respectively.

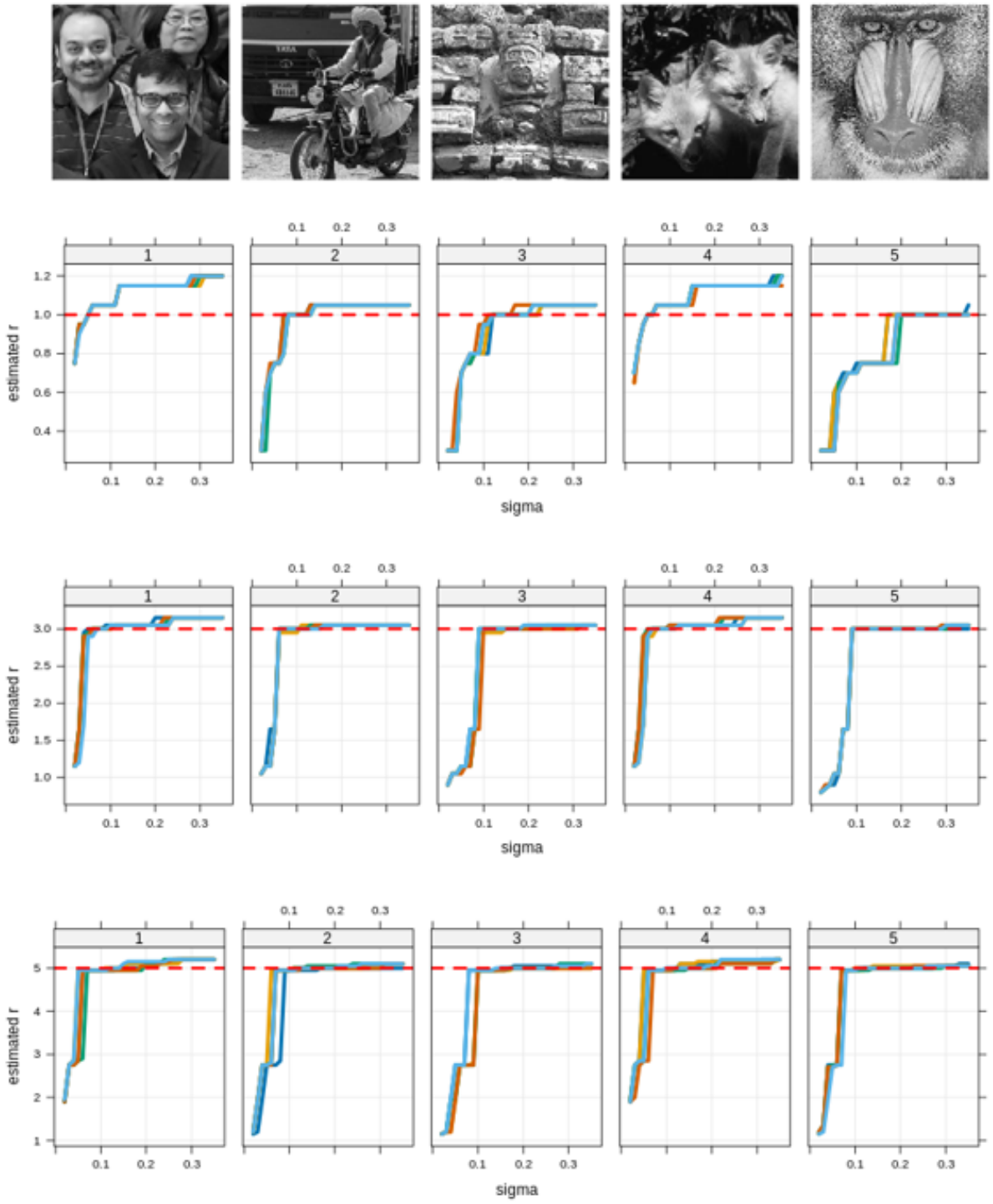

Figure 9: 201 × 201 Patch and Disc Kernel: Top row indicates the sharp images used for simulation and second, third and bottom row corresponds to r_{true} = 1,3,5 respectively.

7 Segment Anything

Discussion on Segment Anything

8 Applying on Segments

Problem with black regions and possible alternatives

9 Decorrelation Loss Function

Motivation and theory

10 Simulated Experiments

Simulated experiment

patch by patch

11 Application on Real Images

Images and Depth map

12 Discussion

Advantages, Limitations

13 Future Work

14 Acknowledgment

References

[1] Brian A. Barsky, Daniel R. Horn, and Klein. "Camera Models and Optical Systems Used in Computer Graphics: Part I, Object-Based Techniques". In: *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2003. URL: http://dx.doi.org/10.1007/3-540-44842-X_26.

[2] P. Grossmann. "Depth from focus". In: *Pattern Recognition Letters* (1987). ISSN: 0167-8655. URL: http://dx.doi.org/10.1016/0167-8655(87)90026-2.

[3] Alexander Kirillov et al. *Segment Anything*. 2023. URL: https://arxiv.org/abs/2304.02643.

[4] Anat Levin et al. "Image and depth from a conventional camera with a coded aperture". In: *ACM transactions on graphics (TOG)* 26.3 (2007), 70–es. URL: http://dx.doi.org/10.1145/1276377.1276464.

[5] Kaustav Nandy. "Locally Dependent Natural Image Priors for Non-blind and Blind Image Deconvolution". PhD thesis. Indian Statistical Institute, 2021. URL: https://digitalcommons.isical.ac.in/doctoral-theses/7/.

[6] Magda Peligrad and Wei Biao Wu. "Central Limit Theorem for Fourier Transforms of Stationary Processes". In: *The Annals of Probability* (2010). ISSN: 0091-1798. URL: http://dx.doi.org/10.1214/10-AOP530.

[7] Deepayan Sarkar and Kaustav Nandy. *rip: Image Processing in R*. New Delhi, India, 2021. URL: https://github.com/deepayan/rip.

[8] Xiang Zhu et al. "Estimating Spatially Varying Defocus Blur From A Single Image". In: (2013). ISSN: 1941-0042. URL: http://dx.doi.org/10.1109/TIP.2013.2279316.