A Method for Refocusing Photos using Depth from Defocus

Benjamin Wiberg
Department of Electrical Engineering
Stanford University
Email: bwiberg@stanford.edu

***Abstract*—This paper will describe a method for refocusing photos taken with a regular camera. By capturing a series of images from the same viewpoint with varying focal depth, a depth metric for the scene can be calculated and used to artificially blur the photo in a realistic way to emulate a new focal depth.**

I. Introduction

The process of refocusing images during post-processing is a powerful tool for photography and video. Professional photographers and cinematographers could employ such methods to be able to adjust the focus after capturing the images, yielding greater artistic control in the post-processing stage.

Focus and blur of a scene from a given viewpoint depends on the depth of the scene, and physical camera blur can be reconstructed using convolution with a blur kernel whose size depends on the depth of that pixel[1]. Therefore a depth image of the scene is needed to perform refocusing. The idea of estimating a depth map of the scene using the focus and defocus characteristics of images is called depth from defocus.

II. Related work

Some work in this field has been done. Most methods rely on multiple images from the same viewpoint, with a varying focal plane. Alex P. Pentland was one of the first to introduce methods for measuring scene depth using inverse filtering [2] based on multiple images. J. Ens et al proposed a matrix-based method for estimating depth and camera parameters [3] with better performance than inverse filtering methods. A recent method proposed by S. Suwajanakorn et al allow for slight variations in rotation and translation of the camera between images [4]. Their method uses optical flow to align the indivudual images. The depth map estimation is done in steps. First, a composite image is created where the entire scene is in focus. This image is used to estimate the amount of defocus or blur for every pixel in every frame, and also a corresponding confidence metric. The blur and confidence maps are used to solve for the camera's aperture and focal length as well as each image's focal depth in conjunction with a depth map of the entire scene.

Some active methods for depth from defocus involve illuminating the scene using some predefined light scheme. F. Moreno-Noguer et al proposed a method lighting the scene with a grid-aligned dot pattern and measuring the defocus spread of each dot [5]. Their approach uses only a single frame in the capturing stage, but needs a set of pre-calibrated images to correctly analyze the dot pattern. Amin, M Junaid et al uses a similar active illumination method to extract depth from a scene [6].

III. Method

The image stack is assumed to be ordered according to their focal depth. The first image is assumed to have its focal plane intersect the objects in the scene where the depth is smallest, i.e. no feature in the scene is closer to the camera than the focal plane of the first image. The last image is assumed to be focused at the most far away objects, i.e. the background objects with the largest depth.

The method for refocusing a stack of images described in this paper can be divided into four different parts. The first part is aligning the images in the focal stack. The second part consists of creating a single image where all parts of the scene are in focus. The third part is to calculate a depth map of the scene using the sharp image and the original image stack. The fourth and last part is the method used to create the actual refocused image.

A. Aligning the images

Since the scale factor of the image decreases as the focal depth increases, a stack images taken from the same viewpoint (e.g. mounted on a tripod) will not be aligned in scale. In addition, the proposed algorithm should handle small translational and rotational differences between images. The alignment is done by aligning all images to a fixed frame, chosen as the image with the nearest focal depth. The reasoning behind this is so that no image will need to be downscaled and therefore show black pixels around the edges.

First, SURF points and features are calculated for each image. For each image that is not the fixed image, a 2D homography matrix is calculated using the RANSAC algorithm on the SURF features of the fixed image and the image to be transformed. Then the target image is transformed according to the resulting homography matrix from the RANSAC algorithm, thus aligning it to the fixed image.

B. Creating the sharp image

An image that is sharp everywhere is created using the images in the stack. This is done by selecting pixels from the images based on the amount of detail, i.e. sharpness, in a local neighbourhood around the pixel. A detail map is calculated as

$I_x^2 + I_y^2$, where I_x is the image convolved with a horizontal Sobel kernel and I_y is the image convolved with a vertical Sobel kernel. The final sharpness measure of each pixel is then calculated as the sum of the detail map's values in a local neighbourhood around the pixel, weighted by a gaussian kernel. Finally, the sharp image is constructed by choosing pixels where this sharpness measure is highest.

C. Calculating the depth map

A depth map of the scene is constructed using the sharp image as well as the first and last images in the image stack (i.e. the ones with near and far focal planes). Since the near-focused image does not have any features closer to the camera than its focal plane, the amount of blur in the image is linearly dependent on the distance from its focal plane. Therefore a blur metric of the first image can be used to approximate the scene depth. The same reasoning can be applied to the far-focused image, but with the reversed relationship between its blurriness and depth.

Blur maps are calculated for both the near-focused and far-focused images. Pre-computation is needed by blurring the sharp image with increasingly larger blur kernels, yielding a new stack of uniformly blurred images. The target image is then compared to this stack of uniformly blurred images in the following way. For each pixel, extract a local neighbourhood in the target image and every blurred image. Calculate the difference between all pixels in the neighbourhood weighted with a gaussian kernel, and select the depth of the pixel as the blur strength of the blurred image that has the lowest difference for that pixel.

A final depth map estimate is calculated by combining the near-focused depth map and the far-focused depth map like $I_{depthnear} + (max(I_{depthfar}) - I_{depthfar}$.

D. Creating the refocused image

Given the sharp image and the depth map, a refocused image can be computed as follows. Given a selected depth d, the desired amount of blur can be calculated as $I_{blur} = |I_{depth} - d|$. The final refocused image is created by convolving the sharp image with a gaussian kernel that has a size that varies spatially based on the blur amount I_{blur}.

IV. Results

The proposed algorithm has been implemented in MATLAB. Refocusing is done based on a focal plane selected by the user.

There is an issue with the sharpness calculation along the edges of the image, as can be seen in figure 1. A manual cropping of the sharp image has therefore been implemented, see figure 2.

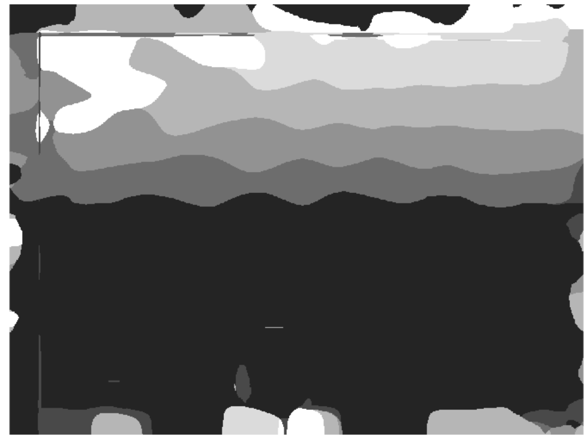

Fig. 1. Sharpness index map, color indicates index in image stack

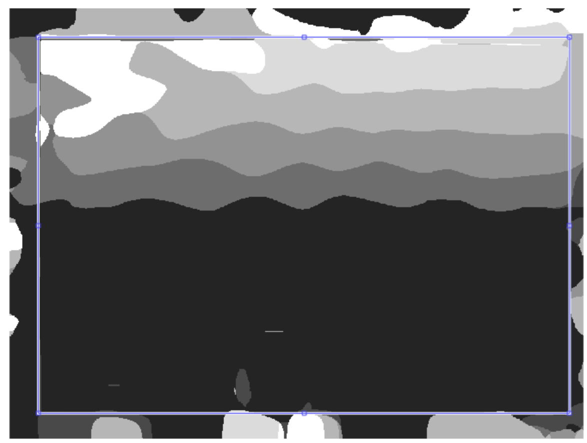

Fig. 2. Manual cropping of faulty edges

A. Test images

The algorithm was tested on a number of image stacks, with different results. The types of scenes that yielded best results were scenes where the depth varied smoothly across the image.

An image stack taken of a laptop keyboard was used. A sample image from the set is shown in figure 3. Its reconstructed depth map is shown in 4. The depth map matches the actual depth alright, since the depth is fairly linearly increasing following the keyboard and the table. An important artifact to discuss is that areas with uniform color and texture in the scene yields poor results in the depth map. This is because the area will never seem detailed, no matter what blur factor is used it will still look roughly the same. For the purposes of refocusing only, this does not make an impact at all, since it does not matter if the depth is off for these pixels since the amount of blur does not change their appearance. A refocusing example of this is shown in figure 5. Compare this with figure 4 for the closest buttons on the keyboard, and note that it does not have any discernible effect on the final refocused image.

Another test set featuring a more complex scene is shown in figure 6. As shown in figure 7, the reconstructed depth map does not seem to correspond to the actual depth in the scene. A refocusing of that image, shown in figure 8, shows a lot of

Fig. 3. Sample image in test image stack of 7 images

Fig. 4. Depth map of keyboard test image stack

blur artifacts that causes the image to look a lot worse.

V. Conclusion

The method described in this paper manages to produce a depth-like map that is good enough for refocusing purposes. In its current state the method does not handle fast variations in depth across the scene well, but with some modifications to the algorithms a better depth map can be calculated.

Acknowledgment

The author would like to thank Jean-Baptiste Boin for counseling during the project, and Gordon Wetzstein for the great lectures and class in general.

Fig. 5. Refocused image of keyboard

Fig. 6. Sample image in test image stack of 21 images

References

[1] Michael Potmesil and Indranil Chakravarty. A lens and aperture camera model for synthetic image generation. *SIGGRAPH Comput. Graph.*, 15(3):297–305, August 1981.

[2] Alex P Pentland. Depth of scene from depth of field. Technical report, DTIC Document, 1982.

[3] J. Ens and P. Lawrence. An investigation of methods for determining depth from focus. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(2):97–108, Feb 1993.

[4] S. Suwajanakorn, C. Hernandez, and S. M. Seitz. Depth from focus with your mobile phone. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3497–3506, June 2015.

[5] Francesc Moreno-Noguer, Peter N Belhumeur, and Shree K Nayar. Active refocusing of images and videos. *ACM Transactions On Graphics (TOG)*, 26(3):67, 2007.

[6] M Junaid Amin and Nabeel A Riza. Motionless active depth from defocus system using smart optics for camera autofocus applications. In *SPIE Photonics Europe*, pages 98960N–98960N. International Society for Optics and Photonics, 2016.

Fig. 7. Depth map of table test image stack

Fig. 8. Refocused image of table