

# We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

**4,800**

Open access books available

**122,000**

International authors and editors

**135M**

Downloads

Our authors are among the

**154**

Countries delivered to

**TOP 1%**

most cited scientists

**12.2%**

Contributors from top 500 universities



**WEB OF SCIENCE™**

Selection of our books indexed in the Book Citation Index  
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?  
Contact [book.department@intechopen.com](mailto:book.department@intechopen.com)

Numbers displayed above are based on latest data collected.

For more information visit [www.intechopen.com](http://www.intechopen.com)



# Estimating 3D Surface Depth Based on Depth-of-Field Image Fusion

Marcin Denkowski<sup>1</sup>, Paweł Mikołajczak<sup>1</sup> and Michał Chlebiej<sup>2</sup>

<sup>1</sup>*Faculty of Computer Science, Maria Curie-Skłodowska University,  
pl. Marii Curie-Skłodowskiej 5, 20-031 Lublin*

<sup>2</sup>*Faculty of Mathematics and Computer Science, Nicolaus Copernicus University,  
Chopina 12/18, 87-100 Toruń  
Poland*

## 1. Introduction

Image fusion is a process of combining a set of images of the same scene into one composite image. The main objective of this technique is to obtain an image that is more suitable for visual perception. This composite image has reduced uncertainty and minimal redundancy while the essential information is maximized. In other words, image fusion integrates redundant and complementary information from multiple images into a composite image but also decreases dimensionality. There are many methods discovered and discussed in literature that focus on image fusion. They vary with the aim of application used, but they can be mainly categorized due to algorithms used into pyramid techniques (Burt (1984); Toet (1989)), morphological methods (Ishita et al. (2006); Mukopadhyay & Chanda (2001); Matsopoulos et al. (1994)), discrete wavelet transform (Li et al. (1995); Chibani & Houacine (2003); Lewiset al. (2007)) and neural network fusion (Ajimmarangsee & Huntsberger (1988)).

The different classification of image fusion involves pixel, feature and symbolic levels (Goshtasby (2007)). Pixel-level algorithms are low level methods and work either in the spatial or in transform domain. This kind of algorithms work as a local operation despite of transform used and can generate undesirable artifacts. These methods can be enhanced by using multiresolution analysis (Burt (1984)) or by complex wavelet transform (Lewiset al. (2007)). Feature-based methods use segmentation algorithms to divide images into relevant patterns and then combine them to create output image by using various properties (Piella (2003)). High-level methods combine image descriptions, typically, in the form of relational graphs (Williams et al. (1999)).

In this work we use image fusion algorithm to achieve first of our aims, i.e. to obtain the deepest possible depth-of-field in macro-photography using standard digital camera images. Macro photography is a type of close-up photography. In the classical definition it is described as photography in which the image on film or electronic sensor is at least as large as the subject. Therefore, on 35mm film, the camera has to have the ability to focus on an area at least as small as  $24 \times 36\text{mm}$ , equivalent to the image size on film (magnification 1:1). In other words, macro photography means photographing objects at extreme close-ups with magnification ratios from about 1:1 to about 10:1. There are some primary difficulties in macro photography; one of the most crucial is the problem of insufficient lighting. When using some

cameras to take photos in the macro-mode, the camera must be positioned so close to the object that it touches the front piece of glass in the lens. In this case it is impossible to place a light source between the camera and the subject, making extreme close-up photography impractical. 50mm is a typical focal-length lens used on a 35mm camera, and can focus so close that the lighting problem remains. The method of choice in such situations is usually to use a telephoto macro lenses. When using such devices in macro photography it is possible to increase the focal length to be greater than 100mm. But this implies second problem of macrophotography – very shallow Depth-of-Field (DOF) (see Figure 1(a)).

Because it is very difficult to obtain high values of DOF for extreme close-ups it is essential to focus on the most important part of the subject. Any other elements that are even a millimeter farther or closer may appear blurred in the acquired photo. For this reason, special devices like advanced tripods for a medium-scale objects or microscope stage for micro-scale objects are required for precise focusing. The depth of field can be defined as the distance in front of and behind the subject appearing in focus. Only a very short range of the photographed subject will appear in exact focus. This focus decreases rapidly on either side of this distance, but due to imperfections of the human eye the focused area seems to be much bigger. This focused area decreases more quickly in front of the focus point than behind as the angle of the light rays changes more rapidly when it is closer to the lens, while becoming parallel with increasing distance. It is for these reasons that there is no precise definition of what is focused; there are many factors that determine whether the subject appears in focus. The most important factor is how a single point is mapped onto the film area. If a given point is exactly at the focus distance it will be imaged as one point on the film, but if this point is farther or nearer it will produce a disk whose border is known as a “circle of confusion”. These circles can be used to define the measure of focus and blurriness as they increase in diameter the further away they are from the focus point. For the most common size of 35mm camera negative (22x16mm), the acceptable “circle of confusion” diameter at which human eye is able to distinguish such a circle as a dot is usually set to 0.05mm. The film size is also important when considering the depth of field problem because, for a given scene, the larger the negative is then the longer the lens needed to capture it. Summarizing, for a specific film format, the depth of field is described as a function parameterized by: the focal length of the lens, the diameter of the lens opening (the aperture), and the distance between the subject and the camera. Let  $D$  be the distance at which the camera is focused,  $F$  the focal length (in millimeters) calculated for an aperture number  $f$  and  $k$  - the “circle of confusion” for a given film format (in millimeters), then depth of field (DOF) (Constant (2000)) can be defined as:

$$DOF_{1,2} = \frac{D}{1 \pm \frac{1000 \times D \times k \times f}{F^2}} \quad (1)$$

where  $DOF_1$  is distance from the camera to the far depth of field limit, and  $DOF_2$  is the distance from the camera to the near depth of field limit. The aperture controls the effective diameter of the lens opening. Reducing the aperture size increases the depth of field, however, it also reduces the amount of light transmitted. Lenses with a short focal length have a greater depth-of-field than long lenses. Greater camera-to-subject distance results in a greater depth-of-field (see Figure 1(b)). We use this optical phenomenon to determine the distances from the camera to every point of the scene which gives as the height map field of this scene. The height map field allows us to achieve our second goal i.e. to create a three-dimensional model of the photographed scene.

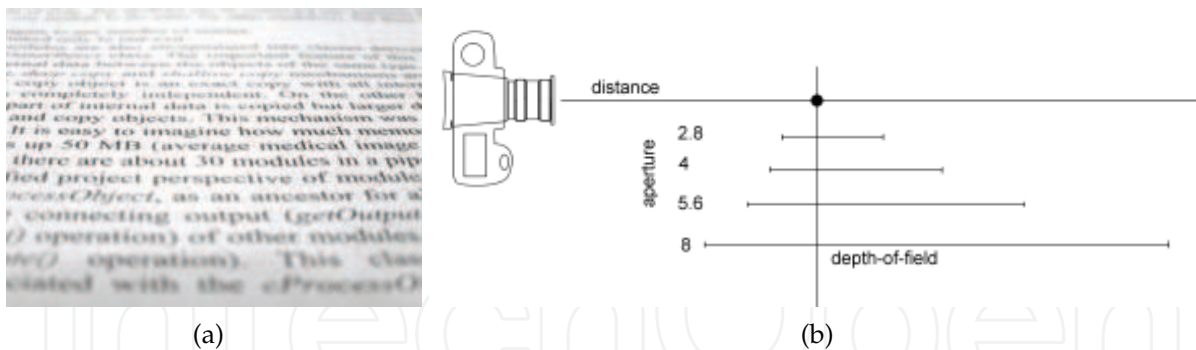


Fig. 1. (a) An example of very shallow depth of field in macro-photography. (b) Example diagram of how the f-number (aperture) affects depth-of-field.

As an input we have created a series of macro photograph images of the same subject with different focus lengths and registered them to each other to create a properly aligned stack of images. The next step was to fuse them into a one composite image. Many of the methods mentioned above can do this perfectly, but our final objective is to create a 3D visualization of that scene. And the main difficulty was to obtain the height map without spikes or noise, generally smooth but with sharp edges. Most of the fusing methods don't care about height map smoothness (if create it at all) because its only goal is to create good fused image. Such an observation determined us to develop a new fusion algorithm.

Our method is based on discrete Fourier transform which copes with problem of height map smoothness. As an effect of fusing algorithm we obtain a height map field and the reconstructed focused image with a very deep depth-of-field. The height map field is a label map which determines the height of each part of the scene. From this map, we can construct a 3D model of the scene.

Generally, we limit our method to macro photography only and we assume that images were taken perpendicularly or almost perpendicularly to the scene. There is also a strong limitation of our method to scenes that can be represented as a height field. The whole method consists of several phases including: image segmentation, height map creation, image reconstruction and 3D scene generation.

## 2. Methodology

We capture our set of images using standard digital SLR camera mounted on a tripod with macro lenses attached. Our method works best when the photographed plan is perpendicular or almost perpendicular to the lens line. It is also good idea to avoid specularities and shining surfaces. For better results gray background can be used. All images are taken in RAW format and then manually calibrated to one another to equalize their illumination, white ballans and exposure.

After that, all images are aligned to each other and the reconstruction process combines the image stack into the height map field and the fused image. We introduce a new method which employs discrete Fourier transform to designate sharp regions in the set of images and combines them together into an image where all regions are properly focused. From the created height map field and the fused image we can generate a 3D surface model of the scene. After that the mesh is created and textured with a plane mapping using the fused image.

The main difficulty is to obtain the height map field without spikes or noise, generally smooth but with sharp edges. It is not essential from the point of view of the image fusion, but it may

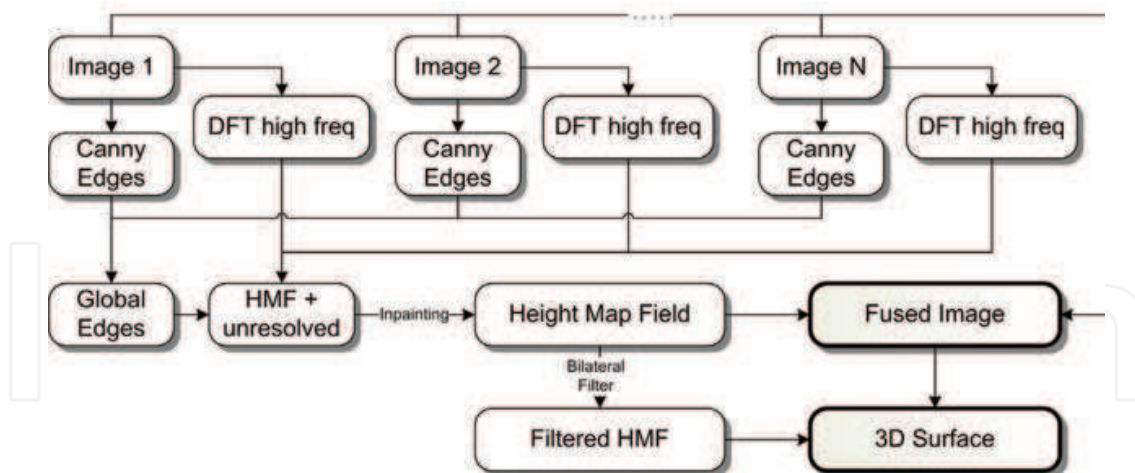


Fig. 2. Image Fusion scheme showing the steps in our method.

be crucial in three-dimensional reconstruction of the scene. Most of such peaks are generated in smooth regions, where noise in defocused region on one image from the stack is often more varied than in the corresponding region on sharp image. This leads to undesired deformations of reconstructed spatial surface. For that reason, we introduced a background plane. For now, we assumed that the background plane overlaps with the last image on the stack, but the user can choose it by hand.

## 2.1 Image fusion

In our work we use discrete Fourier transform methods combined with Canny edge detector and inpainting techniques to distinguish homogeneous regions. Our fusion method is also capable to work with color images. Color image fusion has been discussed in (Bogoni & Hansen (2001)). A naive approach to image fusion in color might include performing image fusion separately and independently on each color plane, then providing the resulting three color planes as a single color image. In practice, this does not work because color in three-dimensional space is a vector and not just three independent components. Similar results gives conversion of RGB images to gray scales and processing them in one-dimensional color space. But in this case a great number of information is lost and it generates another problem: how to map resulted grayscale image back to color space. And our assumption is that the result of the fusion process applied to color images should preserve colors and boundaries between colors. To maximize focus, fusion algorithm must emphasize structural details of the image while the color is preserved. To meet these constraints, from many possible choices for color image representations, we have chosen the CIE  $L^*a^*b$  color space, which separates luminance channel from chromatic channels. LAB also aspires to be perceptually uniform and most complete color space, and its L component closely matches human perception of lightness. Therefore, L channel, which represents the luminance of the color space represent an edge energy itself.

At this stage we assume that images on the image stack are aligned to each other. The main objective is to create the focused image and the height map field (HMF). The whole algorithm diagram is shown in Figure 2.

First, the Discrete Fourier Transform for all images is calculated as follows:

$$F_z(u, v) = \frac{1}{NM} \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} f_z(x, y) e^{-2\pi i (\frac{xu}{N} + \frac{yv}{M})} \tag{2}$$

Where  $N$  and  $M$  are dimensions of the image,  $f_z(x, y)$  is value of the pixel at  $(x, y)$  position taken for  $z - th$  image on the stack. This transform is multiplied with normalized two dimensional Gaussian distribution:

$$F'_z(u, v) = F_z(u, v) * G(u, v) \tag{3}$$

where

$$G(u, v) = \frac{1}{k} \exp \frac{(u + v)^2}{2\sigma^2} \tag{4}$$

where  $k$  is normalization factor and  $\sigma$  is a free parameter determining degree of details preserved, which can be specified by the user. After that inverse transform  $f(x, y) = F^{-1}(u, v)$  is calculated. This gives us an image where pixels with high local gradients are emphasised, see Fig. 5b). This transform is applied to all  $L^*$ ,  $a^*$  and  $b^*$  channels in case of color images, converted previously to CIE  $L^*a^*b^*$  color space. These three channels compose one high frequency map by applying weighted sum operator, e.i.:

$$F'_z(u, v) = 0.8F_z^L(u, v) + 0.1F_z^a(u, v) + 0.1F_z^b(u, v) \tag{5}$$

Next, two metrics: local variance and entropy are calculated for every point of that map:

$$\sigma_z^2(u, v) = \frac{1}{ST} \sum_{j,k}^{S,T} (f_z(u + j, v + k) - \bar{f}_z)^2 \tag{6}$$

$$E_z(u, v) = - \sum_{j,k}^{S,T} f_z(u + j, v + k) \log(f_z(u + j, v + k)) \tag{7}$$

where  $S$  and  $T$  define the size of the local neighbourhood. In our case we use neighbourhood defined as disk-shaped structure with radius equal to  $S/2$ , and  $S = T$ .  $f_z(u, v)$  is a value taken from  $(u, v)$  position in high frequency map  $F'_z$  for  $z - th$  blurry image,  $\bar{f}_z$  is a mean value of the whole neighbourhood for current  $(u, v)$  position.

These two metrics are used for creating height map field according to:

- For every point  $(u, v)$  for both  $\sigma_z^2$  and  $E_z$  square  $y(x) = ax^2 + bx + c$  function is fitted through  $z - th$  dimension.
- Maximum for  $y_\sigma(x)$  and  $y_E(x)$  is calculated and  $x$  position of the maximum ( $Z$ ) is designated as  $Z = 0.5x_{y_\sigma} + 0.5x_{y_E}$
- Height Map Field at  $(u, v)$  position is equal to  $Z$  but only if range of  $\sigma_z^2$  values at  $(u, v)$  position for all  $z$  is greater than  $k$ , otherwise  $HMF(u, v)$  is designated as  $(-1)$ :

$$HMF(u, v) = \begin{cases} Z, & \forall_z \text{range}(\sigma_z^2(u, v)) > k \\ -1, & \text{otherwise} \end{cases} \tag{8}$$

$k$  is a free threshold value controlled by the user. All values equal to  $(-1)$  are treated as unresolved.

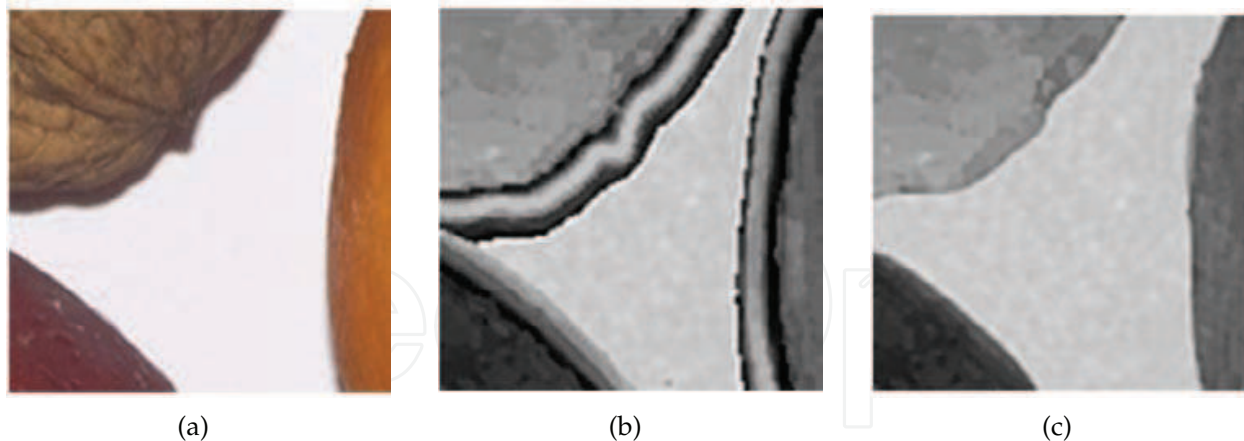


Fig. 3. Example of halo effect: (a) part of the original image, (b) the height map created only based on frequencies - visible halo effect, (c) edges in the height map with help of inpainting.

A base Height Map Field (*HMF*) created by these steps is filtered by hit and miss morphology operator to remove small islands, usually formed because of noise in original images.

The next step is optional but it highly improves the quality of the edges in a resulted image. Figure 3 shows usual problem with halo effect appearing on and nearby edges and where there is a large difference in lightness in local area.

To overcome this problem we use Canny Edge Detector filter which finds sharp edges in all input images. This edge line is locally dilated depending on the strength of the edge in local area. To achieve this we create distance to the edge map for whole edge image and then based on the strength of the edge, they are thickened. The strength of the edge is determined on the basis of difference between original image and its version convolved with a Gaussian filter and the distance map secures that this thickness will only be applied to exact edges. From these edge images we form fused edge image with all edges by simply applying bitwise OR operator for all images. Figure 4 illustrates steps of this edge designating procedure. This process also introduces two free parameters that control its effects: (1) standard deviation of the Gaussian filter and (2) maximum distance to edge. Both parameters are set to default values but much better results are obtained when they are set by the user for a given set of images.

With this edges image we just mark all pixels in a nearby and on the edges in the height map field  $HMF(x,y)$  as unresolved, see Figure 5f).

To classify all unresolved pixels in the height map field *HMF* we distinguish two cases:

1. Field of island formed by linked unresolved pixels in bigger than background factor  $B_f$  - these pixels are marked as background.
2. Otherwise we employ image inpainting technique, described for example in (Bertalmo et al. (2001)) to fill remaining gaps. Inpainting is a technique for reconstructing lost or broken parts of image, widely used for image restoration or noise removing. Generally, the idea is to fill missing gaps using information from the surrounding area. In our work we use Bertalmo algorithm (Bertalmo et al. (2001)) which uses Navier-Stokes partial differential equations with boundary conditions for continuity. An example of inpainting technique is shown in Figure 5g.

Now, we have the height map field prepared to fuse blurry images into a fused one. A value of a fused image pixel  $I_{fused}(x,y)$  is equal to the pixel  $I_i^{(z)}(x,y)$  from  $z$ -th input image on

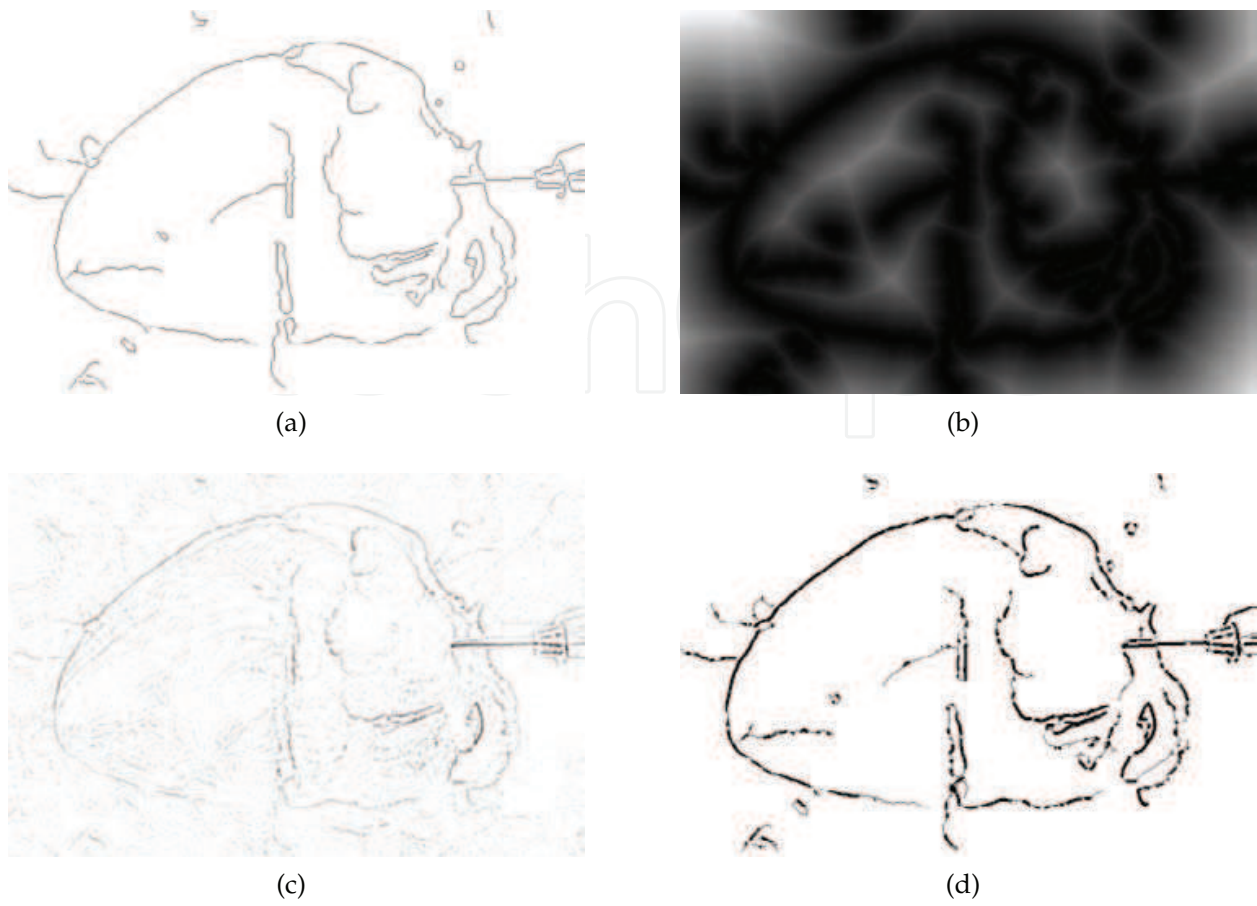


Fig. 4. (a) Canny edges acquired from all blurry images, (b) distance to edge map calculated for canny edge image (scaled in intensity), (c) gaussian strength of the edges (scaled in intensity), (d) final edge map.

the stack, where  $z$  is a value interpolated from the height map field  $HMF(x,y)$ . Separately, regions marked as a background in the  $HMF$  in fused image  $I_{fused}$  are taken from a specific image selected by the user, but generally they can be taken from any image from the stack due to smoothness and not big differences between corresponding images in background regions.

## 2.2 Scene visualization

Before three-dimensional visualization the  $HMF$  is filtered by the median and bilateral filter (Tomasi & Manduchi (1998)) to smooth homogeneous regions while preserving edges between objects (see Figure 8). Bilateral filtering is in details a simple, non-iterative scheme for edge-preserving smoothing, work in spatial and intensity domain and uses shift-invariant low pass Gauss filters. An output pixel's value is calculated according to:

$$h(x) = k \sum_{i \in R} f(x,i) C(x,i) (I(x,i)) \quad (9)$$

where:

$$C(x,i) = \exp \left( -\frac{1}{2} \left( \frac{d(x,i)}{\sigma_d} \right)^2 \right) \quad (10)$$



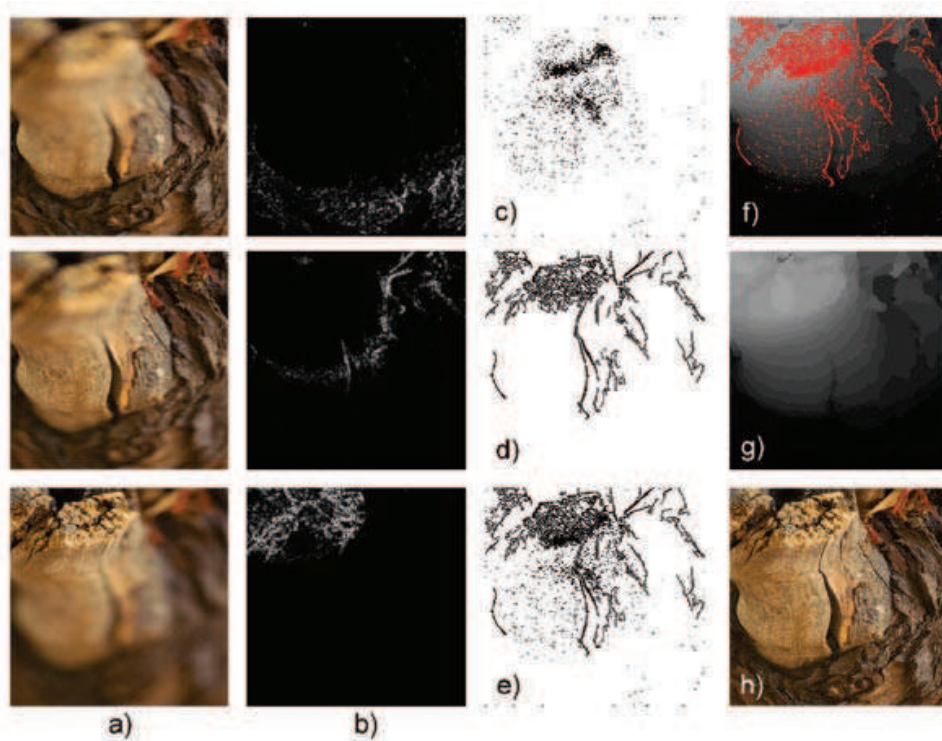


Fig. 5. Example of image fusion in steps; (a) input images; (b) high frequencies in images; (c) unresolved pixels; (d) edges pixels; (e) unresolved + edges pixels; (f) composed Height map field with unresolved pixels; (g) inpainted unresolved pixels in the HMF, (h) fused image.

is a closeness function, a typical Gaussian filter, where  $d(x, i) = d(x - i) = \|x - i\|$  is the Euclidian distance between  $x$  and  $i$ ;

$$I(x, i) = \exp\left(-\frac{1}{2} \left(\frac{\delta(f(x), f(i))}{\sigma_\delta}\right)^2\right) \quad (11)$$

is an intensity function, where  $\delta(\phi, \theta) = \delta(\phi - \theta) = \|\phi - \theta\|$  is a suitable measure of distance between the two intensity values  $\phi$  and  $\theta$ . An examples of different  $\sigma$  parameters for both closeness and intensity functions are shown in Figure 7.

Because the input images are taken from an analogue camera settings, the focus lengths in successive planes do not arrange in a constant or linear function. Thus the user can specify the distances between successive slices in the height map field and HMF is appropriately rescaled in intensities. Now, the HMF is prepared for creating three dimensional surface.

Generally, spatial scene can be visualized by any rendering technique which is able to show information contained in the height map field, where each pixel value represents  $z$  coordinate of appropriate mesh vertex. But, due to very high resolutions of tested images (up to  $4096 \times 4096$ ) a regular triangle mesh (above 16 millions of triangles) can be not the very best choice. Thus, we decided to approximate a height field with an irregular triangle mesh using algorithm similar to (Garland & Heckbert (1995)). The input for this algorithm is a height field map represented by an image whose scalar values are heights and the output is polygonal data consisting of triangles. The algorithm uses a top-down decimation approach and starts with two triangles with vertices positioned at the corners of the height field and, on each pass, locates the point with the greatest error (difference between height field and interpolated mesh approximation) and injects it as a vertex into the mesh using the standard incremental

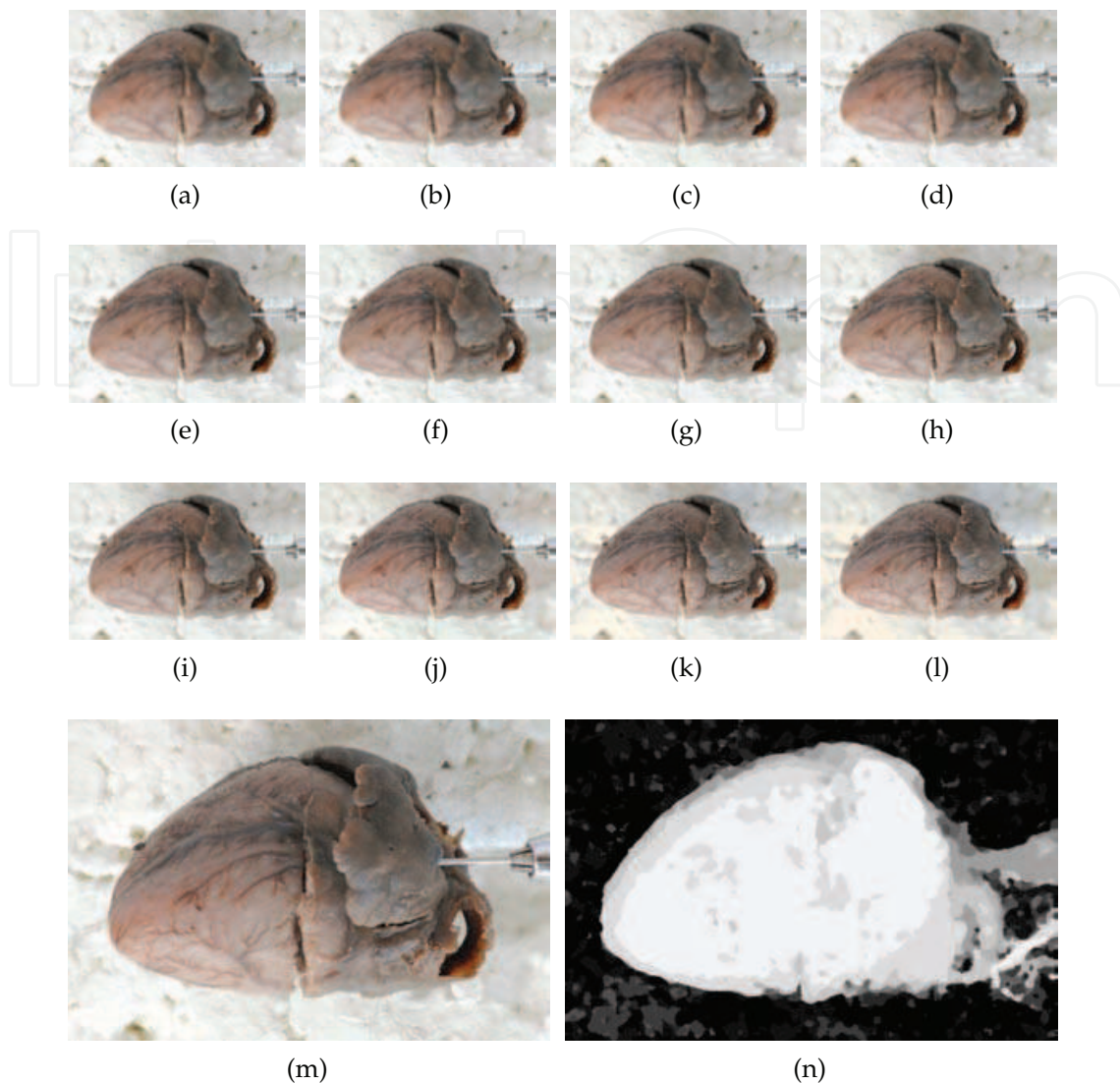


Fig. 6. (a-l) A series of macro photograph taken with different focus length, (m) reconstructed image by our algorithm, (n) the depth map for that series scaled in intensity to fit 8-bit depth. Images courtesy of Department of Anatomy, Medical Faculty, University of Varmia and Masuria in Olsztyn, Poland.

Delaunay point insertion algorithm. The mesh is modified in an iterative fashion until the specific error criterion is met. As a result the number of triangles in the output is reduced as compared to a naive tessellation of the input height field map. From our empirical tests, it seems that the reduction of triangles compared to the regular mesh while preserving good quality vary from 40% to 70% depending on the complexity of the approximated scene. See the differences between regular and irregular mesh in Figure 9.

Generated mesh is smoothed and resulted surface is textured with a plane mapping by the fused image. Additionally, the scene is lit by a directional light which is able to cast shadows to make bumpy surfaces more visible.

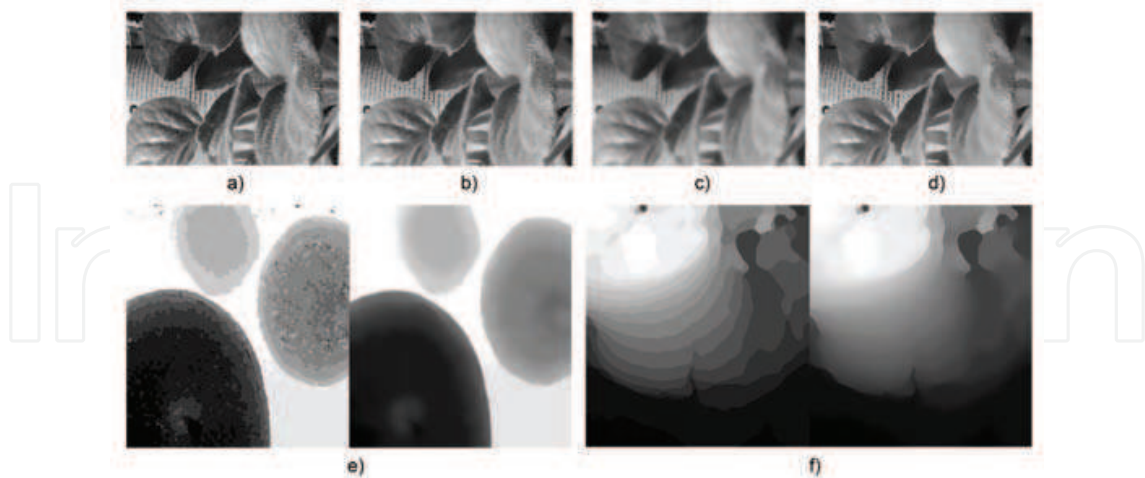


Fig. 7. Example images filtered by bilateral filter; (a) original image; (b) image filtered with radius  $r = 24, \sigma = 20$ ; (c)  $r = 24, \sigma = 96$ ; (d)  $r = 12, \sigma = 12$ , 8 iterations; (e, f) examples of the HMF from fusion algorithm (on left) and the HMF filtered by bilateral filter (on right).

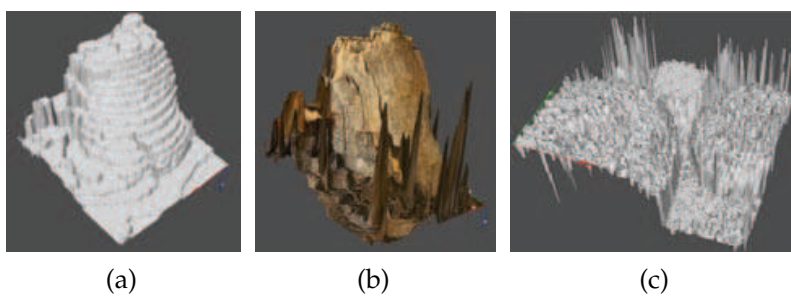


Fig. 8. (a) Surface of 3D model without bilateral smoothing, (b) and (c) examples of 3D model surfaces without removing spikes in fusing algorithm.

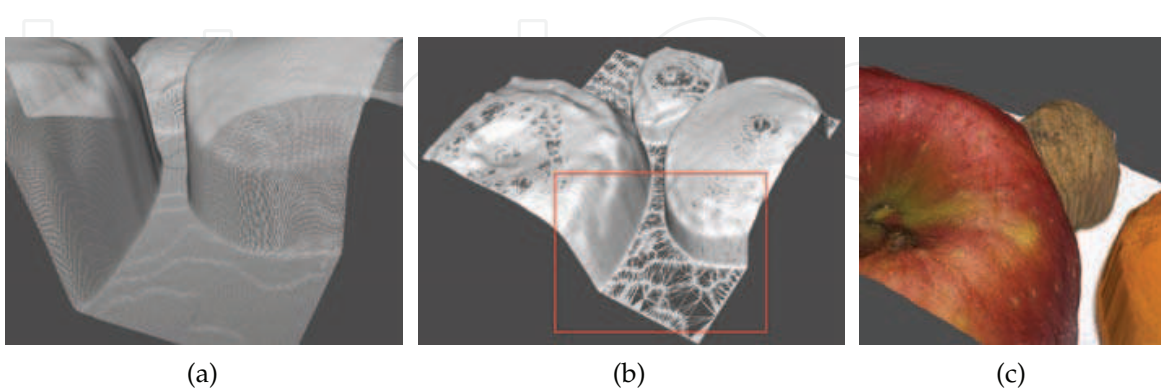


Fig. 9. (a) 3D model generated as a regular triangle mesh, (b) 3D model generated as decimated irregular triangle mesh – red rectangle covers the region visible in (a), (c) full textured 3D model rendered from the same point of view.

### 3. Experimental results

The proposed method has been implemented on Linux platform in C++ language using our Integrated Graphics and Modeling Environment (IGME) framework and Kitware VTK library for visualisation purposes.

To test whole reconstruction procedure we have prepared eight image stacks from macrophotography. Here, each image contains objects at different distances from the camera. Thus, one or more objects naturally become out-of-focus when the image is taken. Each stack contains six to twelve images taken with different depth-of-field. In all cases the procedure is performed in the following order. At first, we have manually equalize all images to each other, then the registration process aligns multifocus images to each other to minimize misregistration. Next, the reconstruction process combine image stack into the height map field and fused image. Finally, 3D scene was generated.

Reconstruction time strongly depends on the size of the images used in the fusion and the number of images on the stack. The fusion process takes about 45%, and generation of three dimensional mesh takes remaining 55% of all time needed for full reconstruction. For a typical set of images, containing ten images with resolution 512x512 the whole procedure lasts about 60 seconds.

#### 3.1 Evaluating image fusion algorithm

Examples of multifocus images with height map fields and reconstructed fused images are shown in Figures 10(a,b) and 11(a,b). Quantitative measure that evaluates the quality of image fusion and produces single numerical score that indicates the success of the fusion process is hard to define and is often performed in impractical subjective trials. We have decided to use a metric  $Q^{AB/F}$  proposed by Xydeas and Petrović in (Xydeas & Petrović (2000)). In this case, a per-pixel measure of information preservation is obtained between each input and the fused image which is aggregated into a single score  $Q^{AB/F}$  using a simple local importance assignment. This metric is based on the assumption that fusion algorithm that transfers input gradient information into result image more accurately performs better. Furthermore, by evaluating the amount of edge information that is transferred from input images to the fused image, a measure of fusion performance can be obtained.  $Q^{AB/F}$  is in range [0,1] where 0 means complete loss of information and 1 means perfect fusion. In our case we have modified this metric to calculate measure of quality for more than two images as it was in case of original  $Q^{AB/F}$  metric:

$$Q^{AB/F} = \frac{\sum_z Q_z^{AF}(n,m)w_z(n,m)}{\sum_z w_z(n,m)} \quad (12)$$

where  $z$  is a number of image on the stack of blurry images and:

$$Q^{AF}(n,m) = Q_g^{AF}(n,m)Q_\alpha^{AF}(n,m) \quad (13)$$

$$Q_g^{AF}(n,m) = \frac{\Gamma_g}{1 + e^{\kappa_g(G^{AF}(n,m) - \sigma_g)}} \quad (14)$$

$$Q_\alpha^{AF}(n,m) = \frac{\Gamma_\alpha}{1 + e^{\kappa_\alpha(A^{AF}(n,m) - \sigma_\alpha)}} \quad (15)$$

where  $A^{AF}(n,m)$  and  $G^{AF}(n,m)$  are defined as in (Xydeas & Petrović (2000)) and describe the relative strength and orientation of the edges in an image using the Sobel operator. Constants

	S1	S2	S3	S4	S5	S6	S7	S8
$Q^{AB/F}$	0.43	0.38	0.53	0.22	0.46	0.39	0.44	0.52

Table 1. The quality measure  $Q^{AB/F}$  for all eight cases.

are:  $\Gamma_g = 0.9994$ ,  $\Gamma_g = 0.9879$ ,  $\kappa_g = -15$ ,  $\kappa_\alpha = -22$ ,  $\sigma_g = 0.5$ ,  $\sigma_\alpha = 0.8$ . Table 1 contains values of  $Q^{AB/F}$  metric that measures quality of image fusion.

### 3.2 Evaluating 3D reconstruction

Figures 10 and 11 show qualitative results of our method for eight tested image sets. The biggest problem in this 3d reconstruction is to obtain a surface which is smooth enough in uniform regions and simultaneously has sharp edges on the objects boundaries. The best results are received when the photographs are taken perpendicularly to the background, objects are within the scene, and they are rough without smooth regions.

Because quantitative measure of the 3D reconstruction of real models is practically impossible we have created synthetic tests. Two simple scenes have been generated in 3D modeling application: (1) a flat surface inclined  $15^\circ$  to the perpendicular plane to camera axis, this surface has been textured and illuminated only with ambient light and (2) spherical surface, also textured and lit with ambient light. For both scenes analytical equation of the surface was known allowing to calculate quantities like volume or shape and to compare with other surface(s). These surfaces are presented in Figure 12.

Both scenes have been rendered with depth-of-field filters that simulates depth-of-field effect using Blender 3D graphic application. In both cases 10 images of partially sharp images have

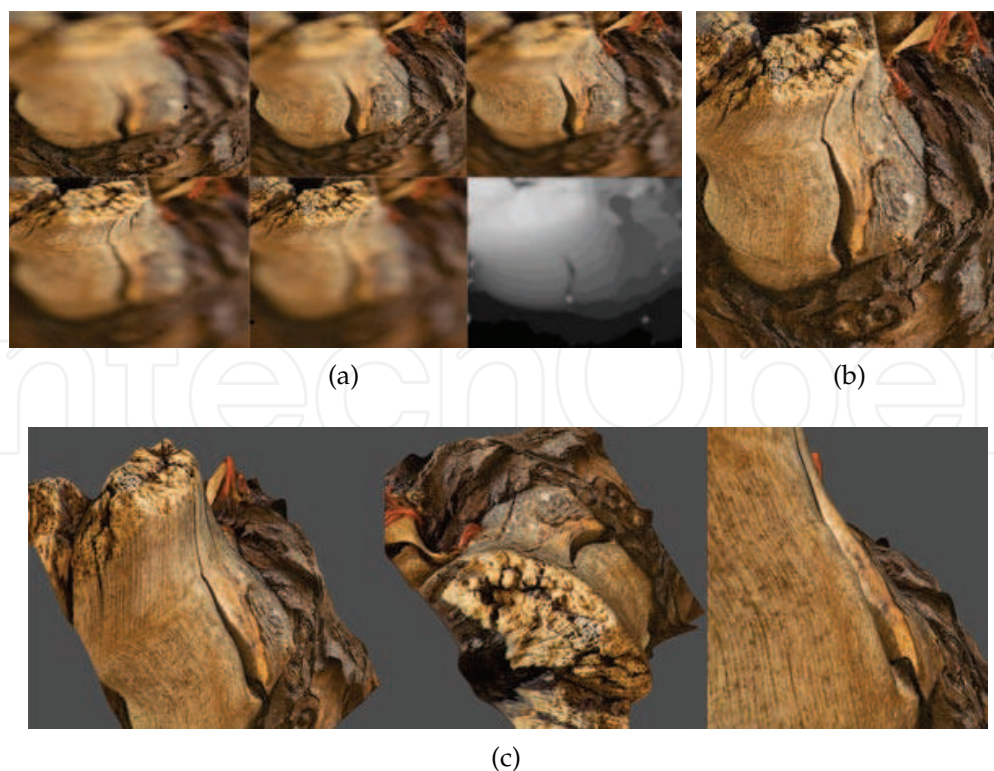


Fig. 10. (a) A few samples of blurry images with height map, (b) final fused image, (c) reconstructed 3D model.



Fig. 11. (a) A few samples of blurry images with height map, (b) finally fused image, (c) reconstructed 3D model.

been created. For every image the focus length was set to different parts of the scene covering a range from the farthest parts of the scene to the nearest ones with constant step between slices. These images were input images to test our fusing and 3D reconstruction method. After reconstruction, created scene was rescaled to match to bounding box of the original scene. We chose two quantities to compare 3D scene generated by 3D application and scene generated by our method:

1. Mean square difference of differences in meshes of reconstructed and original scene:

$$MSD = \frac{1}{R} \sum_{x,y \in R} (f(x,y) - g(x,y))^2 \quad (16)$$

2. Normalized cross-correlation between meshes of reconstructed and original scene:

$$XC = \frac{\sum_{x,y \in R} (f(x,y) - \bar{f}) \cdot (g(x,y) - \bar{g})}{\sqrt{\left( \sum_{x,y \in R} (f(x,y) - \bar{f})^2 \cdot \sum_{x,y \in R} (g(x,y) - \bar{g})^2 \right)}} \quad (17)$$

To calculate both quantities the space of the scene was quantized in  $(X, Y)$  dimension to the size equal to the resolution of the height map field (marked as  $R$ ).  $f(x, y)$  is interpolated height

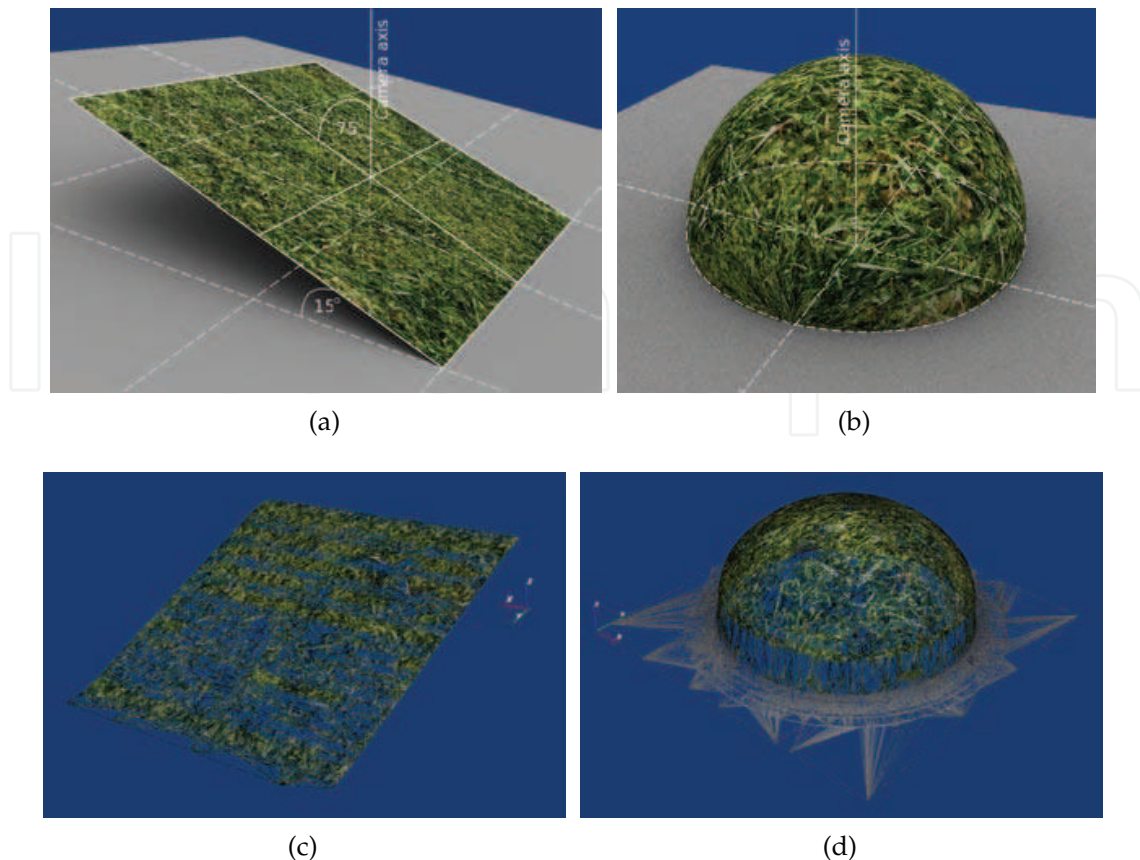


Fig. 12. Two test scenes: (a) flat surface inclined  $15^\circ$  to the perpendicular plane to camera axis, (b) spherical surface, (c) wireframe rendering of the reconstructed plane scene, (d) wireframe rendering of the reconstructed sphere scene.

( $z$  axis) in the ( $x, y$ ) position in reconstructed scene,  $\bar{f}$  is the mean height of reconstructed scene,  $g(x, y)$  is interpolated height in the ( $x, y$ ) position in original scene,  $\bar{g}$  is the mean height of original scene.

Mean square difference estimates the difference between the reconstructed surface and the original one and was equal  $MSD = 0.018$  and  $MSD = 0.031$  respectively for plane and sphere scenes. Cross correlation gives information about how this two surfaces are similar to each other. Two identical surfaces give value of XC equal to 1 or  $(-1)$ , value equal to 0 means completely different surfaces. For plane scene XC was equal to 0.98 and for sphere scene XC was equal to 0.96. Obtained results indicate very good match for both surfaces. However, despite the use of mesh decimation algorithm the number of triangles is still much larger than original mesh. The problem is also “effect of blocks” which must be smoothed by very expensive algorithms for smoothing mesh of triangles. We also expect that synthetic tests would give better results than real tests. To estimate the quality of our procedure we plan to scan a simple but real micro-scene by a 3D laser scanner and then compare with a reconstructed mesh to acquire more precise results.

Figure 13 shows an example of a typical failure. Our method often fails when there are large smooth regions which don't belong to the background plane. The main difficulty in such cases is to distinguish between background and an object without any external spatial knowledge of the scene.

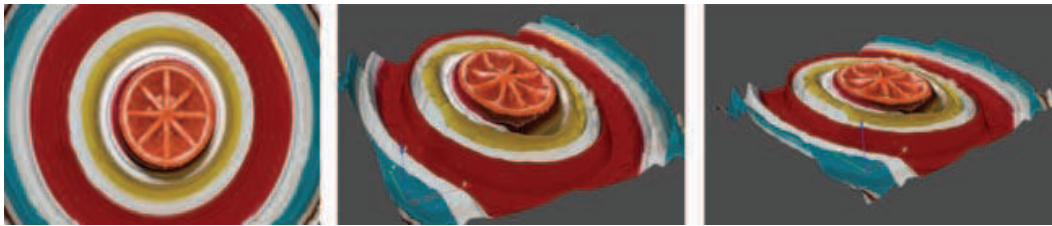


Fig. 13. Typical image that creates failed 3D model. This photograph presents a common child's spinning top. Reconstruction algorithms failed because of many smooth and uniform regions and a lack of background plane.

#### 4. Conclusions

This paper presented a new attempt to the image fusion and estimation of surface depth based on multifocus images. We proposed the whole pipeline from raw photographs to the final spatial surface. Input multifocus images were fused by DFT method and the height map field was created. Based on the *HMF* the image with a greater depth-of-field was composed. Finally, further algorithms reconstructed the 3d surface of the photographed scene.

The presented results of generation of 3D models show that our method is a good tool for acquiring surfaces from a few photographs. However, future work should include automatic detection of the background plane. Second, there should be more complex methods used to identify smooth regions of objects. We think that in both cases pattern recognition algorithms should improve effectiveness of our method. Also Feature-based fusion methods such as (Piella (2003)) could generate more accurate height maps.

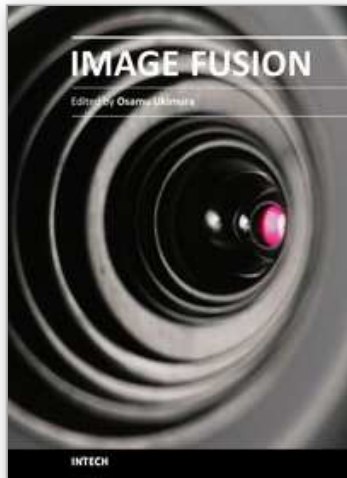
#### 5. References

- Ajjimarangsee, P. & Huntsberger, T. (1988). Neural network model for fusion of visible and infrared sensor outputs, *Sensor Fusion, Spatial Reasoning and Scene Interpretation, The International Society for Optical Engineering*, 1003, SPIE, Bellingham, USA pp. 152-160
- Bertalmo, M., Bertozzi, A. & Sapiro, G. (2001). Navier-Stokes, Fluid Dynamics, and Image and Video Inpainting, *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'01) - Volume 1*
- Bogoni, L. & Hansen, M. (2001). Pattern-selective color image fusion, *Pattern Recognition 34* pp. 1515-1526
- Burt, P. (1984). The pyramid as a structure for efficient computation, *Multiresolution Image Processing and Analysis*, Springer-Verlag, Berlin pp. 6-35
- Chibani, Y. & Houacine, A. (2003). Redundant versus orthogonal wavelet decomposition for multisensor image fusion, *Pattern Recognition 36* pp. 879-887
- Constant, A. (2000). *Close-up Photography*, Butterworth-Heinemann (2000)
- Goshtasby, A. (2007). Guest editorial: Image fusion: Advances in the state of the art, *Information Fusion 8*, pp. 114-118
- Garland, M. & Heckbert, P. (1995). Fast Polygonal Approximations of Terrain and Height Fields, *Technical Report CMU-CS-95-181*, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213
- Ishita, D., Bhabatosh, C. & Buddhajyoti, C. (2006). Enhancing effective depth-of-field by image fusion using mathematical morphology, *Image and Vision Computing 24*, pp. 1278-1287
- Lewis, L., O'Callaghan, R., Nikolov, S., Bull, D. & Canagarajah, N. (2007). Pixel- and



- region-based image fusion with complex wavelets, *Information Fusion* 8, pp. 119-130
- Li, H., Manjunath, H. & Mitra, S. (1995). Multisensor image fusion using the wavelet transform, *Graphical Models and Image Processing* 57 (3), pp. 235-245
- Mukopadhyay, S. & Chanda, B. (2001). Fusion of 2d gray scale images using multiscale morphology, *Pattern Recognition* 34, pp. 1939-1949
- Matsopoulos, G., Marshall, S. & Brunt J. (1994). Multiresolution morphological fusion of mr and ct images of the human brain, *IEEE Proceedings Vision, Image and Signal Processing* 141 (3), pp. 137-142
- Piella, G. (2003). A general framework for multiresolution image fusion: from pixels to regions, *Information Fusion* 4, pp. 259-280
- Toet, A. (1989). Image fusion by rati of low-pass pyramid, *Pattern Recognition Letters* 9 (4), pp. 245-253
- Tomasi, C. & Manduchi, R. (1998). Bilateral Filtering for Gray and Color Images, *Proceedings of the 1998 IEEE International Conference on Computer Vision, Bombay, India*
- Wiliams, M., Wilson, R. & Hancock, E. (1999). Deterministic search for relational graph matching, *Pattern Recognition* 32, pp. 1255-1516
- Xydeas, C. & Petrović, V. (2000). Objective image fusion performance measure, *Electronics Letters* 36 (4), pp. 308-309

IntechOpen



## **Image Fusion**

Edited by Osamu Ukimura

ISBN 978-953-307-679-9

Hard cover, 428 pages

**Publisher** InTech

**Published online** 12, January, 2011

**Published in print edition** January, 2011

Image fusion technology has successfully contributed to various fields such as medical diagnosis and navigation, surveillance systems, remote sensing, digital cameras, military applications, computer vision, etc. Image fusion aims to generate a fused single image which contains more precise reliable visualization of the objects than any source image of them. This book presents various recent advances in research and development in the field of image fusion. It has been created through the diligence and creativity of some of the most accomplished experts in various fields.

### **How to reference**

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Marcin Denkowski, Pawel Mikolajczak and Michal Chlebiej (2011). Estimating 3D Surface Depth Based on Depth-of-Field Image Fusion, Image Fusion, Osamu Ukimura (Ed.), ISBN: 978-953-307-679-9, InTech, Available from: <http://www.intechopen.com/books/image-fusion/estimating-3d-surface-depth-based-on-depth-of-field-image-fusion>

**INTECH**  
open science | open minds

### **InTech Europe**

University Campus STeP Ri  
Slavka Krautzeka 83/A  
51000 Rijeka, Croatia  
Phone: +385 (51) 770 447  
Fax: +385 (51) 686 166  
[www.intechopen.com](http://www.intechopen.com)

### **InTech China**

Unit 405, Office Block, Hotel Equatorial Shanghai  
No.65, Yan An Road (West), Shanghai, 200040, China  
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元  
Phone: +86-21-62489820  
Fax: +86-21-62489821

© 2011 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the [Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License](#), which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.

IntechOpen

IntechOpen