Optimizing depth-of-field based on a range map and a wavelet transform

Mike Wellner[a], Thomas Käster[ab], Thomas Martinetz[b], and Erhardt Barth[b]

[a]Pattern Recognition Company GmbH, Maria-Goeppert Straße 23562 Lübeck, Germany
[b]Institute for Neuro- and Bioinformatics, University of Lübeck, Ratzeburger Allee 160, 23562 Lübeck, Germany

ABSTRACT

The imaging properties of small cameras in mobile devices exclude restricted depth-of-field and range-dependent blur that may provide a sensation of depth. Algorithmic solutions to this problem usually fail because high-quality, dense range maps are hard to obtain, especially with a mobile device. However, methods like stereo, shape from focus stacks, and the use of flashlights may yield coarse and sparse range maps. A standard procedure is to regularize such range maps to make them dense and more accurate. In most cases, regularization leads to insufficient localization, and sharp edges in depth cannot be handled well. In a wavelet basis, an image is defined by its significant wavelet coefficients, only these need to be encoded. If we wish to perform range-dependent image processing, we only need to know the range for the significant wavelet coefficients. We therefore propose a method that determines a sparse range map only for significant wavelet coefficients, then weights the wavelet coefficients depending on the associate range information. The image reconstructed from the resulting wavelet representation exhibits space-variant, range-dependent blur. We present results based on images and range maps obtained with a consumer stereo camera and a stereo mobile phone.

Keywords: Mobile computational photography, stereo cameras, depth-of-field, bokeh, image quality, wavelet transforms, range maps

1. INTRODUCTION

The image quality of small cameras in mobile devices has been improving continuously, but many photographers still buy large cameras instead of using their mobile phones. One important reason is that small cameras do not offer a restricted depth-of-field. Even larger cameras, that do permit significant out-of-focus imaging, often have an unpleasant bokeh, i.e. an unpleasant rendering of out-of-focus regions, compared to cameras with larger sensors and well-designed, fast lenses.[1]

Therefore, a series of attempts have been made to selectively blur images after they have been taken. The probably best known recent application is the SynthCam app created by Marc Levoy.[2] The algorithm is simple and yields good results, but the user needs some practice and must record a video over 15 seconds. Other approaches attempt to segment foreground and background and then blur the background only. Such approaches often yield unpleasant results because imprecise segmentations may lead to artifacts, or because the binary treatment of foreground and background looks unnatural. Images taken with large cameras are nicer not only because the background may be blurred, but because the variation of optical blur as a function of distance generates depth cues and therefore a sensation of depth, a '3D-effect'.

Instead of segmenting foreground and background, one would therefore prefer to use a continuous range map to blur images as a function of the space-variant range information. However, high-resolution range maps are hard to obtain, especially with a mobile device. There are, however, a few techniques, like stereo, shape from

Further author information: (Send correspondence to E.B.)
M.W.: E-mail: mw@prcmail.de
T.K.: E-mail: tk@prcmail.de
T.M.: E-mail: martinetz@inb.uni-luebeck.de
E.B.: E-mail: barth@inb.uni-luebeck.de, Telephone: +49 451 5005503
Published in Proceedings SPIE, Vol. 8667D, Mobile Computational Photography, 2013.

focus stacks, and the use of flash lights which may produce coarse and sparse range maps. A standard procedure is to regularize such range maps to make them dense and more accurate. In most cases, regularization leads to insufficient localization, and sharp edges in depth cannot be handled well. Therefore, photographers still use manual controls and segmentations for selective blurring. Nevertheless, a few attempts have been made to manipulate the depth-of-field without user interventions. A focus stack has been used, for example, by Binder et al.[3] to extend the depth-of-field and by Jacobs et al.[4] to both extend and reduce it. Methods that make use of specialized hardware may provide a much more flexible control of the depth-of-field.[5–9]

In the computer-graphics community similar problems are encountered when images of 3D scenes need to be rendered with realistic camera models. One of the first approaches for rendering with range-dependent blur was due to Potmesil et al.,[10] and various improvements have been proposed (see Barsky et al. for a more recent overview[11]) .

Our approach differs based on the following rationale: often, especially with stereo, the range map is sparse because range can only be estimated given sufficient image structure; but if there is no structure, we must not blur. This argument is not exact because straight edges and periodic structures may not deliver a valid range value but would still have to be blurred; however, problems with uniform regions are more severe and thus the idea seems worth exploring. From the statistics of natural images we know that structured regions are rather rare. This fact is the basis of image compression, which is most effectively implemented based on a wavelet transform as in, e.g., the JPEG 2000 Standard. So, from the perspective of image compression, an image is defined by its significant wavelet coefficients, only these need to be encoded. In other words, if we wish to perform range-dependent image processing, we only need to know the range values for the significant wavelet coefficients. Therefore, the problem of estimating range is very different here from the case where we need range for other purposes like navigation, obstacle avoidance, and 3D measurements.

We therefore propose a method that first computes a wavelet transform, then determines a sparse range map only for the significant wavelet coefficients, and finally weights the wavelet coefficients depending on the associate range information before computing the inverse wavelet transform. As our results demonstrate, the resulting image exhibits space-variant, range-dependent blur, but also some undesired artifacts.

2. HARDWARE

Our work is partially motivated by the fact that, as a rather recent development, low-cost stereo cameras have been launched driven by the interest in 3D stereo displays and the lack, especially on the mass market, of devices that would allow for the creation of 3D content. Notable examples of such devices are the digital camera *Fuji Finepix 3DW3* and the mobile phone *LG P920 3D*. We have used both cameras in our workflow that is described in Fig. 1. The rational was that once such devices are on the market, and in different development pipelines, one should consider also their potential for traditional 2D photography and video. We are therefore somehow misusing these cameras and our results are indeed limited by the fact that the cameras have not been designed for delivering range data. Nevertheless, the results are promising and future devices might as well be optimized for these kind of applications, given that two small cameras will, in general, be smaller, lighter and cheaper than one big camera.

3. ALGORITHM DESCRIPTION

We here describe our algorithm that consists of different modules as shown in Fig. 1. The input to our algorithm is one pair of stereo images denoted by I_L for the image of the left camera and I_R for the image of the right camera. Thus, (I_L, I_R) is our stereo-image pair.

3.1 Camera calibration

In order to compute disparity information from a pair of stereo images, specific stereo-camera parameters have to be known. As an essential preprocessing step, the cameras are calibrated and the camera parameters are used to rectify the image pair and correct for distortions. First, we need to estimate the transformation, i.e., the translation and rotation, between the two stereo cameras. The devices we have used have either fixed camera positions regarding both the disparity and the orientation of the two cameras, or the relative camera positions

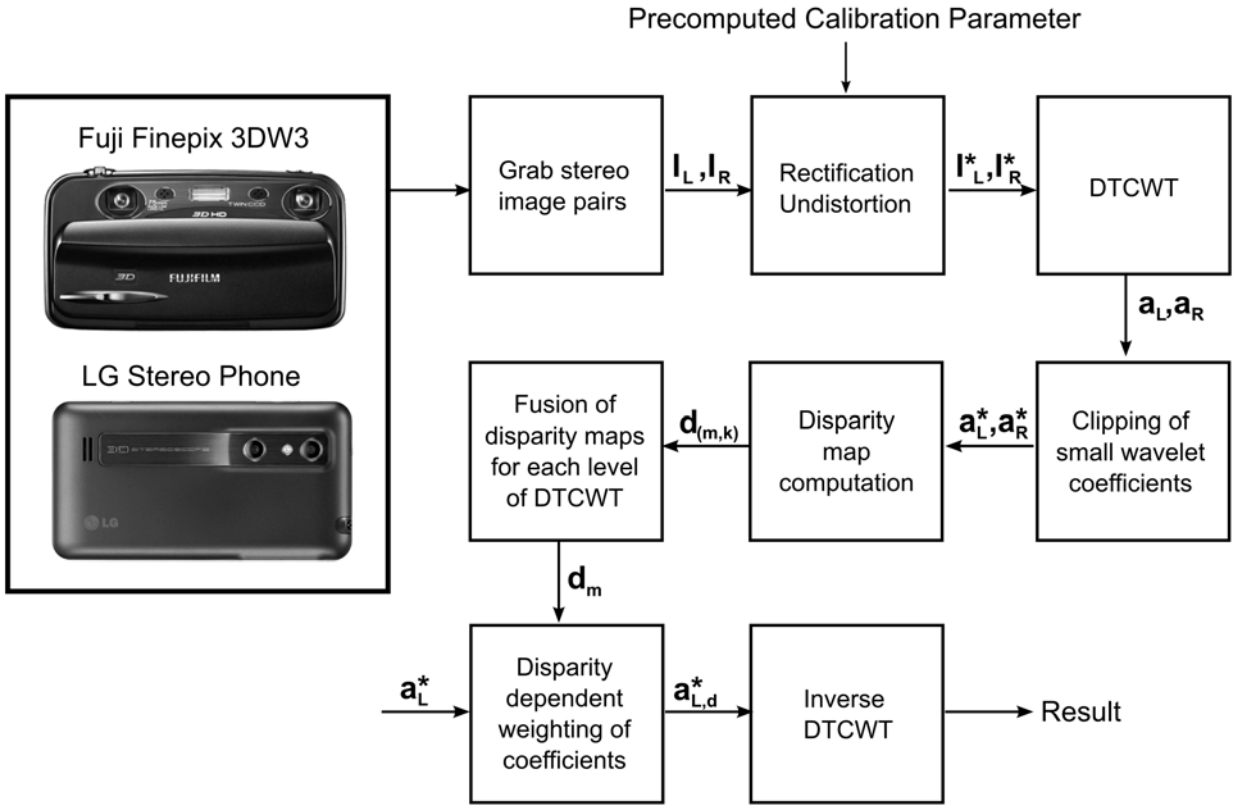


Figure 1. System overview, see text for the description of the modules.

have been fixed to a particular orientation. We used a standard calibration method based on a chessboard pattern and obtained a translation vector T and a rotation matrix $\mathbf{R}$ for the relative camera positions. In addition we computed the distortion vectors D_n for each camera using the known geometry of the chessboard pattern and a set of n stereo image pairs of the chessboard. Finaly, we computed the rectified and undistorted versions of our stereo image pairs (I_L, I_R), which are denoted by (I^*_L, I^*_R).[12]

3.2 Wavelet Transform

As mentioned in the introduction, the range-dependent filtering is performed in the wavelet domain. We have chosen the 2D dual-tree complex wavelet transform (DTCWT) introduced by Kingsbury.[13,14] The wavelet transform is denoted by $a = W(I)$, where W is the wavelet transformation, I is the input image and a are the wavelet coefficients. The wavelet transform has been implemented by using the filtering coefficients introduced by Farras, Selesnick, and Kingsbury.[15,16] The DTCWT is insensitive to shifts,[14,17] it is directionally selective in two and higher dimensions, and has been already applied successfully in computational photography for focus stacking and image fusion.[3]

The wavelet transform W maps any image I to $a_{p \in P} \in \mathbb{C}^{\#P}$, which is a vector of dimension $\#P$ that contains the complex wavelet coefficients. We use $m = 1, ..., M$ to denote the available levels (resolutions) of the transform. The band-pass wavelet coefficients within each scale are enumerated by the sets

$$Q_m = \{m\} \times \{1, 2, 3, 4, 5, 6\} \times S_m \subseteq \mathbb{N}^4, \quad m = 1, ..., M, \tag{1}$$

where $k = 1, ..., 6$ denotes one of six directional sub-bands and S_m is the set of all spatial positions (u, v) within sub-band k of level m. The set $Q = \cup_{m=1}^{M} Q_m$ indexes all wavelet high-pass coefficients. Joining $P = Q \cup R$

we obtain the whole wavelet domain, where $R = S_M$ represents the DC component of the wavelet transform. Finally, Q_{Mag} denotes the magnitude of the complex wavelet coefficient Q.

3.3 Clipping of small wavelet coefficients

According to our strategy outlined above, we would like to operate only on the significant wavelet coefficients. Since in the wavelet domain the informative part of the signal is encoded in large coefficients, we are clipping small coefficients that are below a threshold T:

$$Q_{Mag_{(m,k,u,v)}} = \begin{cases} Q_{Mag_{(m,k,u,v)}}, & \text{if } Q_{Mag_{(m,k,u,v)}} > T, \\ 0, & \text{otherwise .} \end{cases} \tag{2}$$

In our examples, the threshold has been chosen such that 40 percent of the coefficients have been set to zero at each level. Note that such small coefficients are predominantly found in homogeneous image regions. The clipping operation has the additional benefit of denoising the images. The images of these small cameras are rather noisy and the noise may cause problems for the disparity estimation. Also, the significant coefficients in the two images are more similar than the image intensities themselves, since the latter may vary due to differences in exposure and focus setting.

3.4 Disparity estimation

The disparity of the wavelet coefficients in the two images is determined by block matching. The search for the corresponding blocks can be simplified by using the constraints of the stereo setup. After calibration and rectification, we can assume that a template of the left image and the corresponding image block in the right image are on the same horizontal line, i.e., image row. Furthermore, in our examples, we only deal with static objects that are all behind the intersection of the optical axes. We can therefore assume that the disparity has the same sign for all objects in the scene. To obtain a disparity value d for all locations in I_L one has to match a template $B_{(u_L,v_L)}$ in a region $C_{(u_R,v_R)}$ for all center pixels (u_L, v_L), with the additional constraint $u_R \geq u_L$ and $v_R = v_L$. Disparity maps $d_{(m,k)}$ are first computed for each level m and each direction k by using the magnitudes of the wavelet coefficients. Then, maps of the same resolution are fused by computing the average over the different directions resulting in the range maps $d_{(m=1,\ldots,M)}$ for each level. The maps are normalized to the range $[0, 1]$. The disparity is set to zero for the clipped coefficients, which have zero magnitude. Since there is no structure in the image at these positions it should not matter whether we blur the image or not.

3.5 Manipulating the depth-of-field in the wavelet domain

Given the wavelet coefficients Q_{Mag} of $a_L = W(I_L)$ of the left input image and the disparity maps $d_{(1,\ldots,M)}$ for each resolution $m \in M$ we compute the new, weighted coefficients Q^*_{Mag} as

$$Q^*_{Mag_{(m,k,u,v)}} = Q_{Mag_{(m,k,u,v)}} \times (1 - d_{(m,u,v)}), \tag{3}$$

i.e., we simply multiply the coefficients by the normalized and inverted disparity map. Alternative weighting schemes may be used, especially after gaining more insight into the perceptual issues related to the quality of the blur and its desired dependence on range.

Note that objects in the foreground do not change their position relative to I_L and I_R and therefore have a disparity $d_{(m,u,v)} = 0$, which keeps their coefficients unchanged. Objects in the background have a disparity $d_{(m,u,v)} > 0$ which results in a reduction of the corresponding coefficients. This is due to the way the two cameras are adjusted: they are tilted towards each other and therefore their optical axes cross as shown in Fig. 2. The gray square in Fig. 2 is in front of the point of intersection and therefore seems to move to the left. The black square is located beyond the point of intersection and therefore seems to move to the right. In the example images we took, all objects were behind the point of intersection.

3.6 Image reconstruction

Images are reconstructed from the manipulated wavelet coefficients simply by performing the inverse wavelet transform.

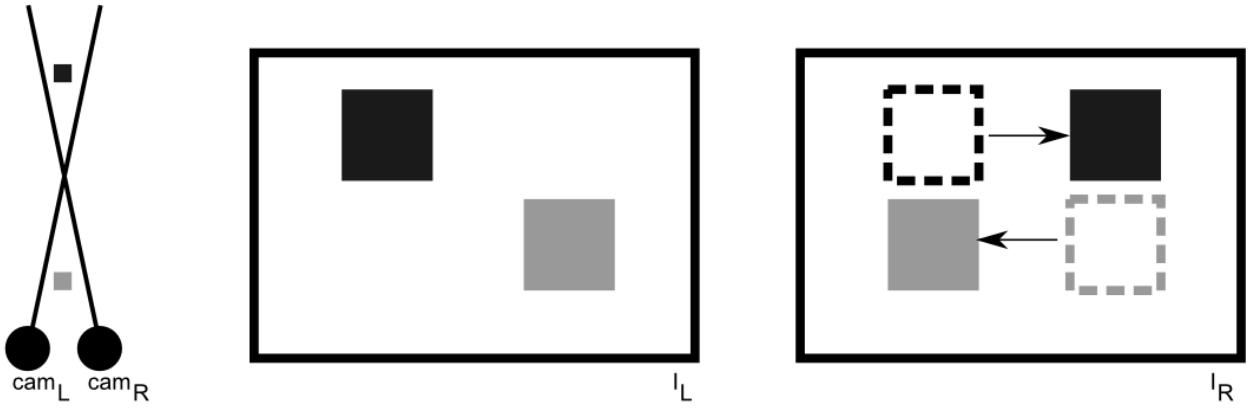


Figure 2. Schematic stereo-camera setup and resulting stereo images of the shown scene. See text for details.

4. RESULTS

The results presented in Figures 3 and 4 use the same format and differ only in the subject and the camera: the subject in Fig. 3 (Lisa) has been photographed by one of the authors with the Fuji camera and the subject in Fig. 4 (Schwalbe) with the LG phone. The top two images (a) are the stereo pair as it comes out of the cameras. The left image below (b) is the left rectified and cropped image, which is input to the algorithm. To simplify computations, the images have been converted to grey-scale images. This image and the corresponding right image are used to estimate the disparities of their wavelet coefficients. The wavelet coefficients of the left image are then weighted by the normalized disparity. The resulting image, obtained with the inverse wavelet transform of the weighted coefficients, is shown on the right (c). The inverted disparity maps of the wavelet coefficients at different levels and orientations of the wavelet transform are shown at the bottom (d).

In print, the visible effect of filtering in the resulting images might be small due to the small size of the images. When enlarging the images in the electronic version of the paper, however, the effect should be apparent. Note that the blur effect as such is natural and pleasing. It can be further adjusted by changing the weighting function in Eq. 3 and the number of scales that are used. Also note, however, that there are a number of artifacts, mainly due to the fact that certain parts of the background are not blurred. This effect is more pronounced in those image regions which have been occluded in the right image and can therefore be observed at the left borders of the foreground objects. Note that in case of the Lisa image, the effect also appears in non-occluded regions due to the repetitive nature of the background, which causes false disparity matches.

5. DISCUSSION

We have presented an efficient method for producing shallower depth-of-field 2D images with low-cost 3D stereo cameras. The method can be integrated with image compression since it operates on the wavelet coefficients of the stereo-image pair. The preliminary results are promising. They are presented without any post-processing and contain some undesired artifacts. The major problem is that for those parts of the background, which are occluded in one of the stereo images, we cannot obtain accurate disparity values and thus the corresponding wavelet coefficients cannot be modulated correctly. As a consequence, some parts of the background will not be blurred. Depending on the background, this may be more or less of a problem. In the computer vision and graphics literature, there a number of methods, some cited above, for improving the quality of a range map by segmenting and filling-in occluded regions. We have not systematically explored these options yet. We have, however, obtained good results with additional segmentation methods, like flash-no-flash exposures and multilinear filtering, which are beyond the scope of this paper. A major problem with the further development of these methods is that we lack a good perceptual model of perceived image defocus and the associated human preferences. Nevertheless, the method presented here can become a useful algorithmic component for mobile computational photography and the design of compact cameras with a pleasing balance of image focus and defocus. An additional benefit is that blurred regions, i.e. reduced wavelet coefficients, can be further compressed,

thus reducing the worldwide quite large amount of stored clutter, such as the one in the background of the Lisa image.

ACKNOWLEDGMENTS

This work has been partially supported by the *German Federal Ministry of Education and Research* within the *KMU Innovativ* program. The grant number is 01IS10009.

REFERENCES

[1] Nasse, H., "Depth of field and bokeh," *Carl Zeiss Camera Lens Division Report* (2010).

[2] Levoy, M., "Synthcam app for the iphone 4, 3gs, ipod touch 4g, and ipad2," (2011).

[3] Binder, T., Kriener, F., Wichner, C., Wille, M., Wellner, M., Kaester, T., Martinetz, T., and Barth, E., "How to make a small phone camera shoot like a big dslr: creating and fusing multi-modal exposure series," in [*Human Vision and Electronic Imaging*], *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series* **8291** (2012).

[4] Jacobs, D. E., Baek, J., and Levoy, M., "Focal stack compositing for depth of field control," *Stanford Computer Graphics Laboratory Technical Report* **2012-1** (2012).

[5] Raskar, R., Tumblin, J., Mohan, A., Agrawal, A., and Li, Y., "Computational Photography ," in [*Eurographics state of the art report*], 1–20 (2006).

[6] Ng, R., Levoy, M., Brdif, M., and Duval, G., "Light field photography with a hand-held plenoptic camera," *Stanford Computer Graphics Laboratory Technical Report* **2005-2** (2005).

[7] Kuthirummal, S., Nagahara, H., Zhou, C., and Nayar, S. K., "Flexible depth of field photography," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **33**(1), 58–71 (2011).

[8] Perwass, C. and Wietzke, L., "Single lens 3d-camera with extended depth-of-field," in [*Human Vision and Electronic Imaging*], *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series* **8291** (2012).

[9] Kolb, A., Barth, E., Koch, R., and Larsen, R., "Time-of-Flight Sensors in Computer Graphics," *Eurographics State of the Art Reports* , 119–134 (2009).

[10] Potmesil, M. and Chakravarty, I., "Synthetic image generation with a lens and aperture camera model," *ACM Trans. Graph.* **1**, 85–108 (Apr. 1982).

[11] Barsky, B. A. and Kosloff, T. J., "Algorithms for rendering depth of field effects in computer graphics," in [*Proceedings of the 12th WSEAS international conference on Computers*], *ICCOMP'08*, 999–1010, World Scientific and Engineering Academy and Society (WSEAS), Stevens Point, Wisconsin, USA (2008).

[12] Bouguet, J. Y., "Camera calibration toolbox for matlab," (2004).

[13] Ivan W. Selesnick, R. G. and Kingsbury, N. G., "The dual-tree complex wavelet transform," in [*IEEE Signal Processing Mag*], 1053–5888 (Nov 2005).

[14] N.G.Kingsbury, "The dual-tree complex wavelet transform. a new technique for shift invariance and directional filters," in [*8th IEEE DSP Workshop*], (Utah, Aug. 9-12 1998).

[15] Abdelnour, A. F. and Selesnick, I. W., "Nearly symmetric orthogonal wavelet bases," in [*IEEE Int. Conf. Acoust., Speech, Signal Processing (ICASSP)*], (May 2001).

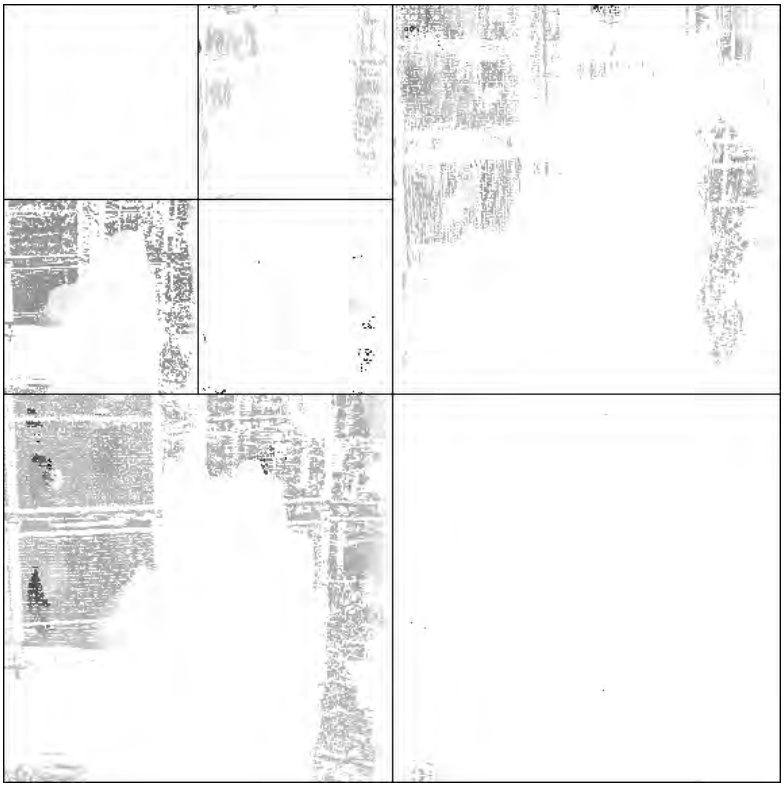
[16] Kingsbury, N., "Image processing with complex wavelets," *Phil. Trans. Royal Society London A* **357**, 2543–2560 (1997).

[17] N.G.Kingsbury, "Complex wavelet for shift invariant analysis and filtering of signals," in [*Appl. Comput. Harmon Anal., vol. 10*], 234–253, (May 2001).

(a)

(b)

(c)



(d)

Figure 3. Results obtained with the Fuji camera: (a) original stereo images; (b) rectified, corrected, and cropped black-and-white image I_L; (c) result after inverse-DTCWT; (d) disparity map for the different levels and directions (see text).

(a)

(b)

(c)

(d)

Figure 4. Results obtained with the LG phone: (a) original stereo images; (b) rectified, corrected, and cropped black-and-white image I_L; (c) result after inverse-DTCWT; (d) disparity map for the different levels and directions (see text).