HAL
open science

Focus mismatch detection in stereoscopic content

Frédéric Devernay, Sergi Pujades, Vijay Ch.A.V.

To cite this version:

Frédéric Devernay, Sergi Pujades, Vijay Ch.A.V.. Focus mismatch detection in stereoscopic content. Stereoscopic Displays and Applications, Jan 2012, San Francisco, United States. hal-00654115v1

HAL Id: hal-00654115

https://inria.hal.science/hal-00654115v1

Submitted on 20 Dec 2011 (v1), last revised 24 Dec 2011 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Focus mismatch detection in stereoscopic content

Frédéric Devernay, Sergi Pujades and Vijay Ch.A.V.

INRIA Grenoble - Rhône-Alpes, France

ABSTRACT

Live-action stereoscopic content production requires a stereo rig with two cameras precisely matched and aligned. While most deviations from this perfect setup can be corrected either live or in post-production, a difference in the focus distance or focus range between the two cameras will lead to unrecoverable degradations of the stereoscopic footage. In this paper we detect focus mismatch between views of a stereoscopic pair in four steps. First, we compute a dense disparity map. Then, we use a measure to compare focus in both images. After this, we use robust statistics to find which images zones have a different focus. Finally, to give useful feedback, we show the results on the original images and give hints on how to solve the focus mismatch.

Keywords: Stereoscopic cinema, Focus mismatch, Depth from focus, Depth from defocus.

1. INTRODUCTION

In this paper, we introduce a method to detect if a stereoscopic pair of images is shot with cameras having the same focus distance and depth of field.

Live-action stereoscopic content production requires a stereo rig with two cameras precisely matched and aligned. This is known to be an on-set time-consuming operation. As opposed to most deviations that can be corrected either live or in post-production, a difference in the focus distance or focus range between the two cameras will lead to unrecoverable degradations of the stereoscopic footage. Identifying focus mismatch is thus an important issue. Whether this is a source of visual fatigue or not is still disputable: most studies are on the fatigue caused by the fusion of a blurred image with a non-blurred image,[1–4] but the effect of having the same amount of blur in both images with different focus distances is still to be looked upon. In 3D-TV and 3D cinema, focus also plays an artistic role, and most cinematographers strongly believe that a focus mismatch can interfere with the 3D effect.[5]

We want to detect the focus mismatch using only the information provided by the captured images, and we assume that we don't have any information about the lens used and their actual settings, such as focal length, focus and iris. Although there exist standard protocols to retrieve this information from the lenses, not all the lens comply with those protocols (especially in cinema) and the provided data is most of the time not accurate enough. This inaccuracy mainly appears because the lens is usually factory-calibrated and the actual values depend on their position with respect to the image sensor.

Measuring the focal blur from an image is an ill-posed problem. For instance, if we have a textureless area in a scene, an in-focus image and the out-of-focus image of this same area will look alike. Similarly, an in-focus image of a poster showing a blurred photograph will appear blurred. However, in our case, we are not interested in measuring the focal blur itself: what we want to detect is a focus *difference* between the left and right views, since it may be disturbing or cause visual fatigue when viewing the stereoscopic images.

In this work, we do not intend to restore the in-focus images from the out-of-focus images by image processing techniques: our goal is only to detect if there is a difference in focus distance or depth of field, using fast enough computations to do it while shooting the images. Moreover, we want to give hints to the photographer or cinematographer pointing the nature of the problem, such as which camera is focusing closer than the other, and if the depth-of-field is the same for both cameras, so that the problem can be fixed immediately.

E-mail: `frederic.devernay@inria.fr`. This work was done within the 3DLive project supported by the French Ministry of Industry `http://3dlive-project.com/`

F. Devernay, S. Pujades and V. Ch.A.V., "Focus mismatch detection in stereoscopic content", Stereoscopic Displays & Applications XXIII, Woods, Holliman, Favalora (eds.), *Proc. SPIE* 8288, paper 8288-12, 2012.
© 2012 Society of Photo-Optical Instrumentation Engineers. Systematic or multiple reproduction, distribution, duplication of any material in this paper for a fee or for commercial purposes, or modification of the content of the paper are prohibited.

2. FOCAL BLUR MODEL, FOCAL BLUR DIFFERENCE AND IMAGE BLUR

2.1 Focal Blur Model

The image blur is often modeled as the convolution of the perfectly focused images with a point spread function (PSF). In our case, we suppose that the PSF is only caused by the focal blur, and neglect other optical aberrations such as diffraction effects. Although various models for the PSF have been proposed in the vision literature,[6,7] the effect of defocus is most often modeled with a 2-D Gaussian PSF.[8–11]

If field curvature can be neglected, the focal blur size σ depends on the depth, taken as the distance from the imaged 3-D point to the focal plane (not to be mistaken with the distance to the optical center). When two cameras are in stereo configuration, the optical axes are near-parallel, and they are orthogonal to the baseline joining the two optical centers. In this case, the definition of depth is the same for both cameras, and the stereo disparity only depends on depth. Moreover, it has been shown[12,13] that the focal blur size is linear with the stereo disparity, as shown in Fig. 1. The focal blur model has thus only two parameters: the focus distance (FD) and the depth of field (DOF). We can consider that elements of the scene at a disparity with a focal blur size smaller or equal to some fixed threshold are in-focus, while elements at a disparity with a bigger focal blur are out-of-focus.

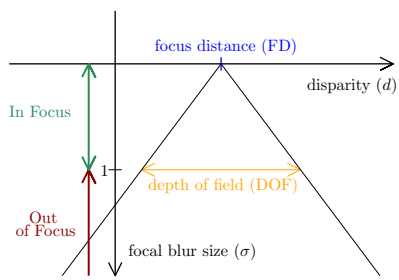

Figure 1. The focal blur model: the focal blur size as a function of stereo disparity is piecewise linear. It is 0 at the focal distance, and the slope is the inverse of the depth of field (DOF).

2.2 Focal Blur Difference

From the focal blur function for each camera, we consider the difference between the two blur functions. Each parameter, respectively FD and DOF, falls into 3 categories: it may be less in the left image than in the right image, equal in both images, or greater in the left image than in the right image. A global shift or scaling on these parameters will not change the shape of the curve. If we put together these 3 cases for each of the two parameters (FD and DOF) we get the 3×3 table from Fig. 2.

2.3 Image Blur

In the two previous subsections we have considered the focal blur model. In our approach we do not have access to the camera settings, but only to the captured images. This means that we cannot measure the focal blur directly, but only image blur. Despite the fact that we cannot measure the focal blur from the images themselves, one of the key points of our method is that the *image* blur difference and the *focal* blur difference should have the same sign at each pixel. Consequently, we use the sign of the image blur difference to detect a focal blur mismatch. Fig. 3 shows, in addition to Fig. 2, the curves corresponding to the sign of the focal blur difference (in orange): it is the catalog of admissible shapes for the graph of the sign of the image blur difference.

Notice that if we only consider the shape of the sign curves of Fig. 3, we are not able to distinguish between the cases ($\mathrm{DOF}_l < \mathrm{DOF}_r$, $\mathrm{FD}_l < \mathrm{FD}_r$) and ($\mathrm{DOF}_l < \mathrm{DOF}_r$, $\mathrm{FD}_l > \mathrm{FD}_r$). Similarly ($\mathrm{DOF}_l > \mathrm{DOF}_r$, $\mathrm{FD}_l < \mathrm{FD}_r$) has the same shape as ($\mathrm{DOF}_l > \mathrm{DOF}_r$, $\mathrm{FD}_l < \mathrm{FD}_r$). In Sec. 3.5, we will discuss how do we still manage to get useful information in those cases.

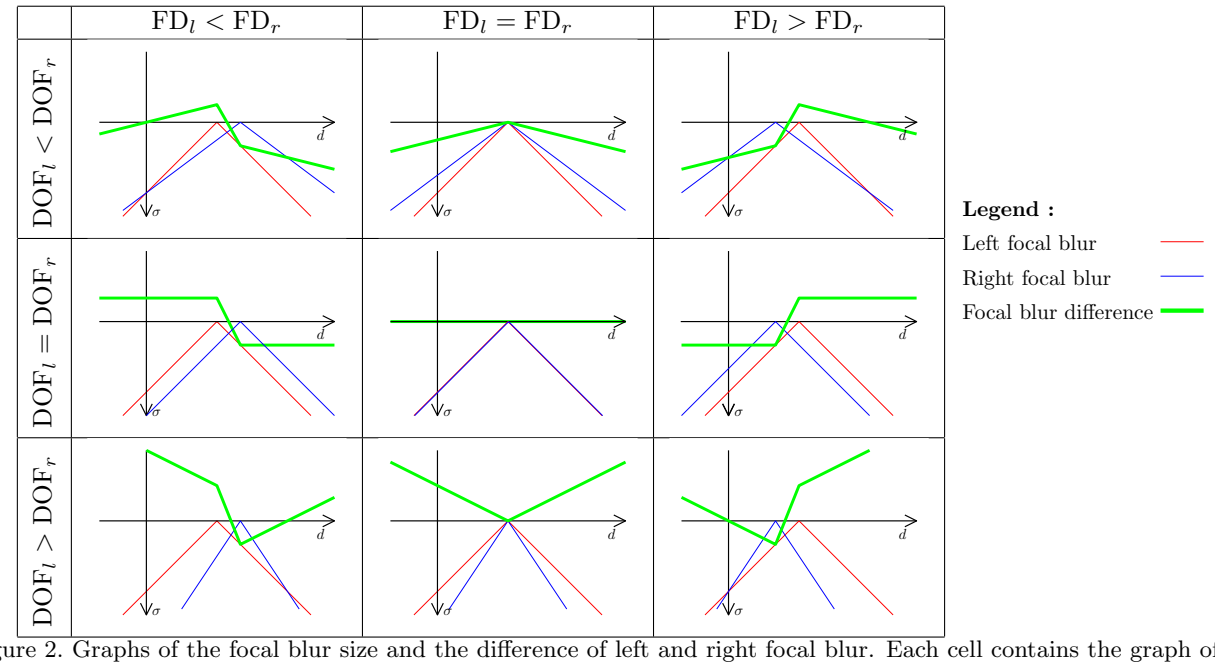

Figure 2. Graphs of the focal blur size and the difference of left and right focal blur. Each cell contains the graph of the left and right focal blur functions, in red and blue respectively, corresponding to the given parameters. The difference of both functions is shown in green.

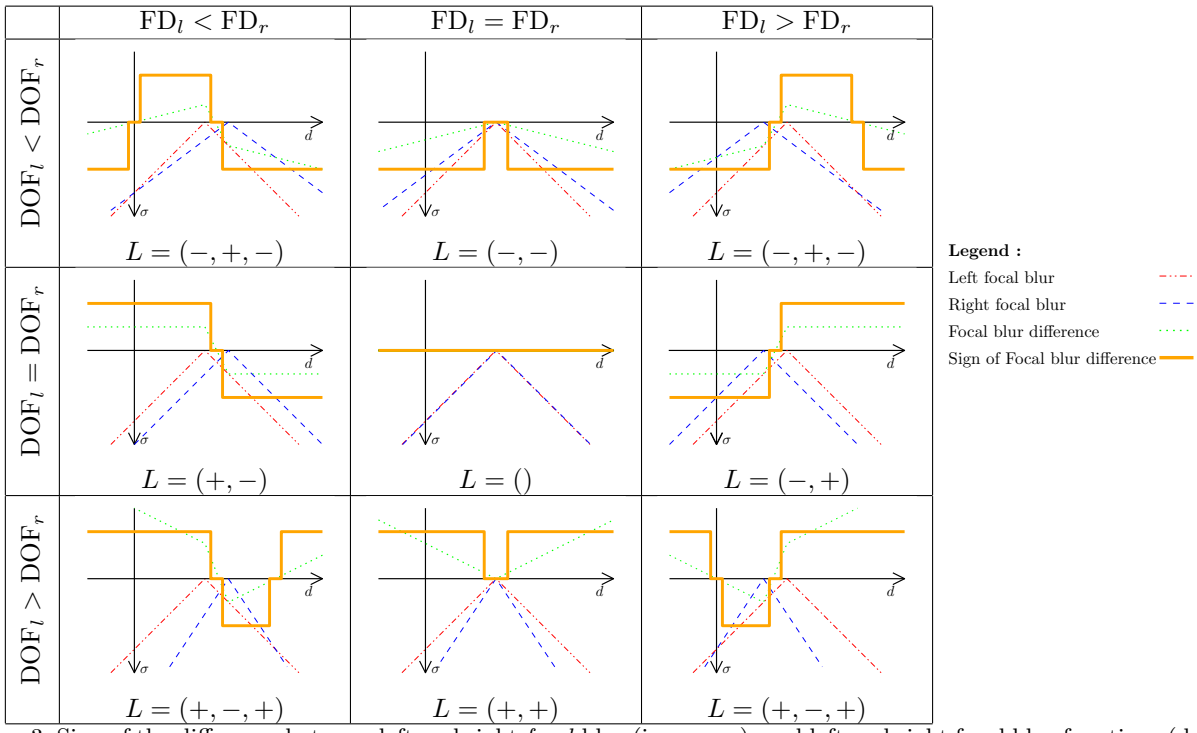

Figure 3. Sign of the difference between left and right *focal* blur (in orange), and left and right focal blur functions (dashed red and blue lines). The sign of the difference between the left and right *image* blur should give the same result. For each shape, we also give the list L of signs used in the classification stage (see Sec. 3.5).

3. FOCUS MISMATCH MEASUREMENT

The algorithm that computes the focus mismatch between the left and right images, in terms of focus distance and depth-of-field, is as follows. We first compute a dense disparity map to find the relationship between the pixels of the left image and the right image. Then we measure the image blur on each image and the sign of its difference for each pair of corresponding pixels from the left and right images (the correspondence function comes from the disparity map). This gives us noisy data onto which we fit a focal blur difference model. Finally we try to extract useful information from this model and give feedback to the operator. The steps that compose the algorithm are thus:

1. Compute a dense disparity map between the left and right images.
2. Measure the image blur difference at each pixel, and aggregate for each disparity value.
3. Extract a model from the focus mismatch measures.
4. Classify the extracted model, in order to get hints on how to solve the problem.
5. Give feedback to the operator.

3.1 Disparity Map Computation

First, we need to find a stereo disparity map which relates the left and the right images. Stereo disparity computation itself is a very active research topic in computer vision, and recent advances showed that, in most cases, a very accurate disparity map can be computed in a reasonable time (even sometimes at video-rate). However, difficult situations such as reduced depth-of-field, low-texture areas, depth discontinuities, repetitive patterns, transparencies or specular reflections are still very challenging and cause local errors in most disparity computation methods.

We can assume that the original images were rectified, so that there is no vertical disparity between the two images: epipolar lines are horizontal, and a point (x_l, y) in the left image corresponds to a point at the same y coordinate (x_r, y) in the right image.

For our approach we need a disparity map computation which preforms well even with reduced depth-of-field and out of focus images. We thus use a real-time multi-scale method[14] which finds good disparity values even between focused and blurred textures. We also compute semi-occluded areas by left-right consistency check, and ignore them in the following computation. Let $d(\mathbf{i})$ be the left-to-right disparity of pixel $\mathbf{i} = (x_l, y)$: pixel $\mathbf{i}$ corresponds to pixel $(x_r, y) = (x_l + d(\mathbf{i}), y)$ in the right image.

3.2 Image Blur Measurement

In order to compare the image blur of corresponding points from the left and right images we use the SML operator (sum of modified Laplacian)[15,16] which was primarily designed for depth-from-focus applications:

$$\begin{aligned}\mathrm{SML}(\mathbf{i}) = &\sum_{i=x-N}^{x+N}\sum_{j=y-N}^{y+N} \nabla^2_{ML} I(i,j), \text{ for } \nabla^2_{ML} I(i,j) \geq T, \\ &\text{where } \nabla^2_{ML} I(x,y) = |2I(x,y) - I(x-s,y) - I(x+s,y)| + |2I(x,y) - I(x,y-s) - I(x,y+s)| \,. \quad (1)\end{aligned}$$

T is a discrimination threshold value, N is the window size used for the computation of SML, and s is the step size. In our experiment, these were set respectively to $T = 5$ (for 8-bits images), $N = 1$ and $s = 1$. For a pixel $\mathbf{i} = (x, y)$ in the *left* image, $\mathrm{SML}_l(\mathbf{i})$ is the SML operator computed at this pixel, and $\mathrm{SML}_r(\mathbf{i})$ is computed at the corresponding pixel in the *right* image.

Let $M(\mathbf{i})$ be the sign of the difference of SML between two pixels:

$$M(\mathbf{i}) = \mathrm{sign}\left(\mathrm{SML}_l(\mathbf{i}) - \mathrm{SML}_r(\mathbf{i})\right). \quad (2)$$

It is positive if left is more focused than right, and negative if right is more focused than left.

Let $w(\mathbf{i})$ be the max of $\mathrm{SML}_l(\mathbf{i})$ and $\mathrm{SML}_r(\mathbf{i})$:

$$w(\mathbf{i}) = \max\left(|\mathrm{SML}_l(\mathbf{i})|, |\mathrm{SML}_r(\mathbf{i})|\right). \tag{3}$$

It is high if one image is textured and in focus, which tells us how reliable the image blur difference information is.

For each disparity value of the scene, $M(d)$ is the mean of $M(\mathbf{i})$ for pixels which have disparity d, weighted by $w(\mathbf{i})$, and $w(d)$ is the corresponding sum of weights. This gives us an estimate of which image is more focused at disparity d.

However this estimate may be very noisy, mainly for one of the following reasons:

- The disparity information may be inaccurate, causing $M(\mathbf{i})$ to be wrong.
- Focal blur of foreground objects may bleed over depth discontinuities : background objects may be wrongly measured as blurred.
- The number of pixel measures available for each disparity depends highly on scene content and the information may even be missing for some disparity values.

Although the measures are noisy, the sign of the image blur difference as a function of disparity is constrained to have one of the shapes presented Fig. 3. The next step consists in enforcing this constraint.

3.3 Model Estimation from the Sign of Image Blur Difference

Given the measures $M(d)$ and $w(d)$, we fit a function $C(d)$ minimizing the energy functional

$$E(C) = \sum_d w(d)E_{\mathrm{data}}(d) + \lambda E_{\mathrm{smooth}}(d), \text{ with } E_{\mathrm{data}}(d) = M(d) - C(d) \text{ and } E_{\mathrm{smooth}}(d) = |C(d-1) - C(d)|. \tag{4}$$

For each disparity d, the function $C(d)$ may represent one of the three following states: "Left is more focused than right", "Right is more focused than left" and "Both images have the same focus", which correspond to the values $+1, 0, -1$. In our effective computations, we want to be robust to noise, so we allow $C(d)$ to take 5 possible values:

- Right more in focus (-0.7)
- Right maybe more in focus (-0.3)
- Same Focus (0)
- Left maybe more in focus (0.3)
- Left more in focus (0.7)

The fact that the sign of the focus difference must correspond to one of the nine shapes from Fig. 3 brings one more constraint to the model: the number of sign transitions is limited. The number of sign transitions are counted as one (1) from a positive or negative value to zero, and as two (2) from a positive to a negative value or from a negative value to a positive value. Observing the shapes from Fig. 3 we can note that the maximum number of allowed sign transitions for $C(d)$ is four (4).

However this constraint is necessary but not sufficient. Some curves may have 4 or less sign transitions but do not correspond to any curve from Fig. 3, or not even to a partial section of them (further discussion on incomplete models in Sec. 3.5.2). Fig. 4 shows the shapes of the forbidden curves. These forbidden solutions are pruned during the optimization.

This optimization problem can be solved in linear time ($O(d_{\max} - d_{\min})$) using dynamic programming. The result is a curve telling for each disparity if left image is more focused than right image, both focus are the same, or right image is more focused than left image.

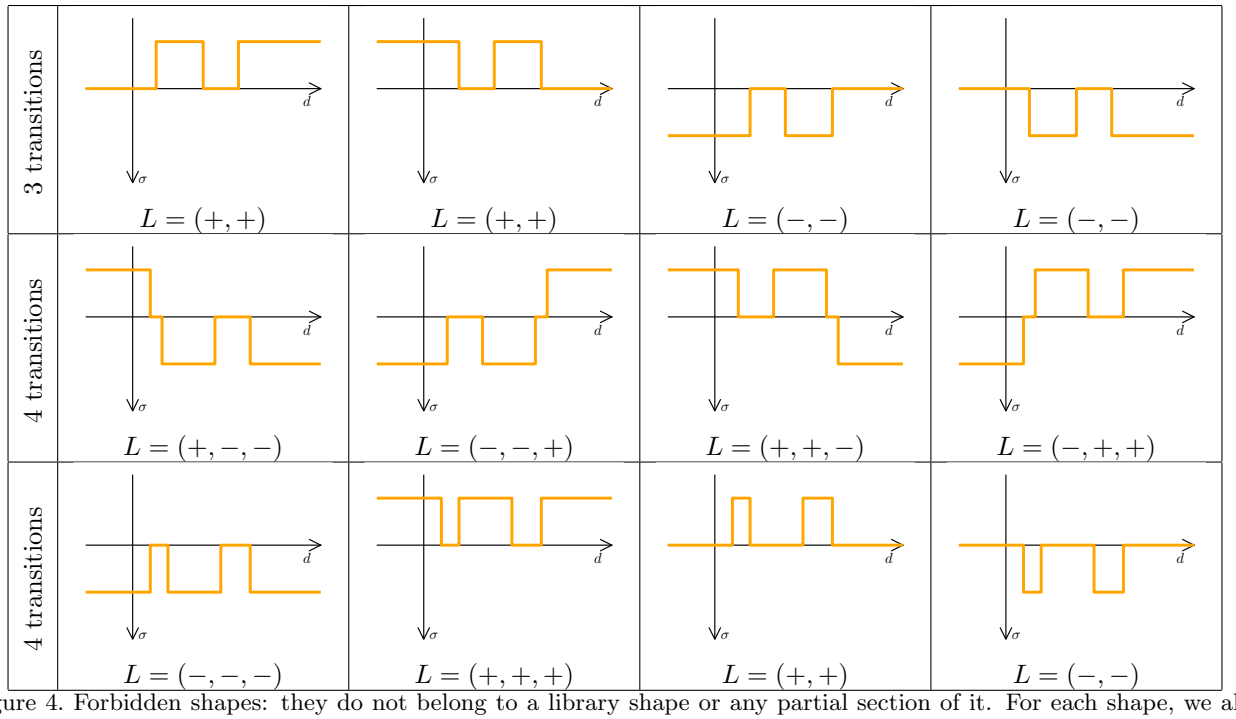

Figure 4. Forbidden shapes: they do not belong to a library shape or any partial section of it. For each shape, we also give the list L of signs used in the classification stage (see Sec. 3.5).

3.4 Visualization of Focus Mismatch

At this stage, we can already visualize the raw results of focus mismatch computation between the images, and as we will see later, this may even be the only output of the algorithm in the most difficult cases. We do this drawing zebra stripes in each image if the image is less focused in that area. The first option is to do this per-pixel, using the function $M(\mathbf{i})$ from Eq. 2, but we found out that the result was too noisy and difficult to interpret (see Fig. 10 for an example).

A better visualization can be obtained by using the function $C(d)$ computed previously: since we have the disparity d at each pixel, we can draw zebra stripes according to the output of the $C(d)$ function: single zebras where this image may be less in focus than the other (i.e. $C(d) = -0.3$ if we are drawing on the left image), and double zebras where we are certain that this image is less in focus than the other (i.e. $C(d) = -0.7$ if we are drawing on the left image). That way, we avoid displaying the raw noise present in the $M(\mathbf{i})$ function, and the resulting images are easily interpretable (Fig. 12). This already gives visual information on which camera should be modified, and the operator may be left to interpret visually the cause of this focus mismatch.

In most practical cases, especially if there are enough measures around the focus distances of both cameras, we are also able to classify the function $C(d)$ and to give a more precise answer about what action is required to fix the focus mismatch problem.

3.5 Classification of the Sign of Image Blur Difference Function

We now want to classify the function $C(d)$ into one of the cases shown in the shape catalog (Fig. 3). To do so, we build a list L of positive and negative parts in the function, sorted by increasing disparity. Fig. 3 and Fig. 4 give, for each authorized or forbidden function shape, the corresponding list L. Since at most 4 sign transitions are allowed for the function $C(d)$, the list has between 0 and 3 elements.

3.5.1 Assuming Measures are Available for all Disparity Values

If we had reliable measures for *all the disparity range*, corresponding to depths from zero to infinity, then we could easily classify the function ($C(d)$ using this list L and Fig. 3. Since we pruned forbidden configurations in

the previous step, the only possible configurations for L are:

$$L \in \{(), (+), (-), (+,+), (-,-), (+,-), (-,+), (+,-,+), (-,+,-)\}. \tag{5}$$

Each of these configurations, except $L = (+)$ and $L = (-)$, can be matched with those from Fig. 3, and gives indications about the cause of the focus mismatch.

However, the assumption that we have reliable measures for the whole disparity range is quite unrealistic, and we have to make weaker assumptions to be able to use this method in practice.

3.5.2 A More Realistic Case: Sparse Measures

Depending on the scene geometry, we may only have information at a few disparity values. This gives us a partial shape of the full focus difference model. In many cases, partial shapes of two different functions may give the same measures, making it difficult to choose which focus mismatch generated these measures.

Here are two different examples, showing how difficult it may be to estimate the shape of the focus difference function from partial measures:

Example 1: $\forall d \in [\mathrm{d_{min}}, \mathrm{d_{max}}]$, $C(d) = 1$ and $L = (+)$. All possible candidates for the model are shown in Fig. 5.

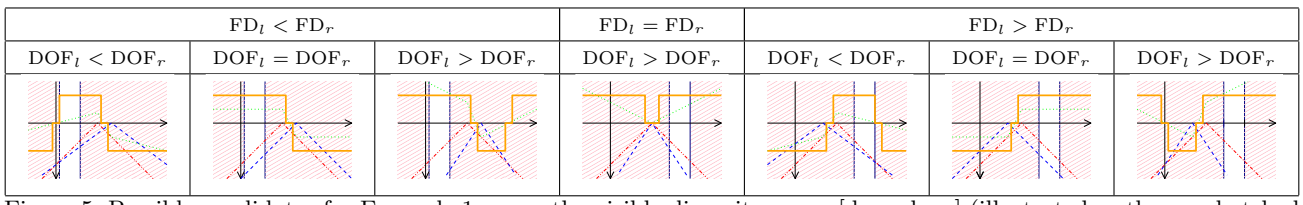

Figure 5. Possible candidates for Example 1 : over the visible disparity range $[\mathrm{d_{min}}, \mathrm{d_{max}}]$ (illustrated as the non-hatched interval), $C(d) = 1$. There are 7 possible configurations of the relative focus. However, supposing that the in-focus distance is within the visible range, this form of $C(d)$ should never be observed.

Example 2: Let $C(d) = \begin{cases} +1 & \text{if } d \leq d_0, \\ -1 & \text{if } d > d_0. \end{cases}$ and $L = (+,-)$. All possible candidates for the model are shown in Fig. 6.

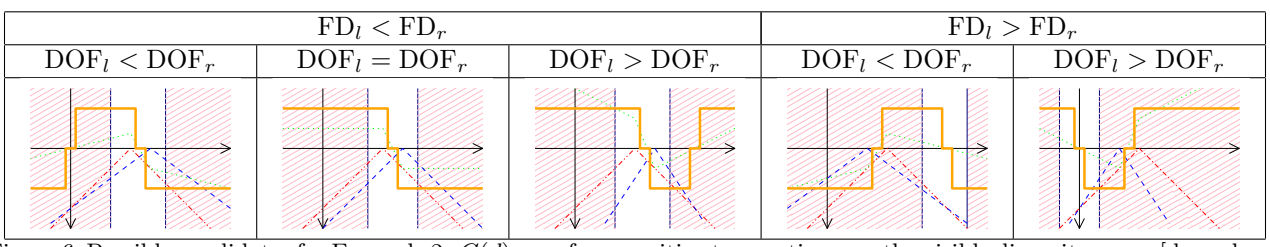

Figure 6. Possible candidates for Example 2: $C(d)$ goes from positive to negative over the visible disparity range $[\mathrm{d_{min}}, \mathrm{d_{max}}]$ (illustrated as the non-hatched interval). There are 5 possible configurations of the relative focus. However, supposing that the in-focus distance is within the visible range, only the configurations from Fig. 7 can be observed.

As we see in Fig. 5 and Fig. 6, a lot of candidates models match our $C(d)$ function. In order to reduce the number of possible candidates, a reasonable assumption seems to consider that the operator has done the focus of one camera on an object of the scene, and that the focus distance of the second camera is within the observable disparity range. This reduces the number of possible candidates for our model. For the second example this leaves us with the candidates of Fig. 7, which share the common feature that $\mathrm{FD}_l < \mathrm{FD}_r$.

As a matter of fact, it can be easily seen that if there are disparity measures for the two focus distances of the cameras, then we can *always* answer to the question: *which focus distance is closer than the other?* Once the

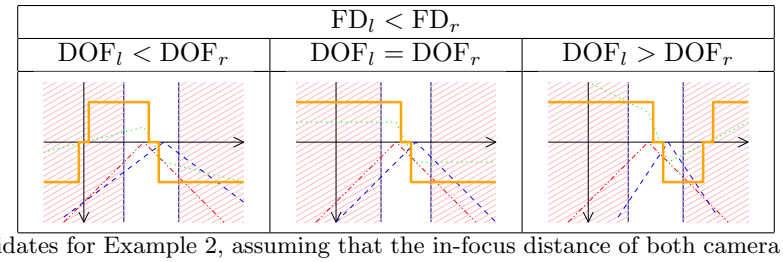

Figure 7. Possible candidates for Example 2, assuming that the in-focus distance of both camera is withing the measured disparity range. Only 3 configurations remain from the 5 in Fig. 6.

operator has fixed the focus distance of both cameras so that they match ($FD_l = FD_r$), we can *always* answer to the question: *which depth-of-field is larger than the other?*

4. RESULTS

The results section is divided in three parts. In the first one we describe how we generated the test images to validate our approach. In the second one we describe the details of the algorithm outline with one concrete example. In the third one we analyze the global results obtained with all the generated test images.

4.1 Ray-traced Synthetic Images

In order to validate our approach we created a set of test images. To have full control over the camera parameters we used a ray tracer tool and generated images of the "Patio" scene by Jaime Vives Piqueres (`http://www.ignorancia.org`). We generated left and right images with different focus distances and different depth of field. For each left and right view we generated five images, that we will call $\{L, R\}$All, $\{L, R\}$Near, $\{L, R\}$Near2, $\{L, R\}$Far and $\{L, R\}$Far2. $\{L, R\}$All images have an infinite depth of field. $\{L, R\}$Far images have a focus distance at disparity $d_{\text{Far}} = 14.2$ (green dot on Fig. 10.a) and $\{L, R\}$Near images have a focus distance at disparity $d_{\text{Near}} = 10.2$ (blue dot on Fig. 10.a). $\{L, R\}$Far and $\{L, R\}$Near images have an aperture value of 5, creating a finite depth of field. $\{L, R\}$Far2 and $\{L, R\}$Near2 images have an aperture value of 10 (twice as big as $\{L, R\}$Far and $\{L, R\}$Near), creating thus a smaller depth of field. All images are shown in Fig. 8.

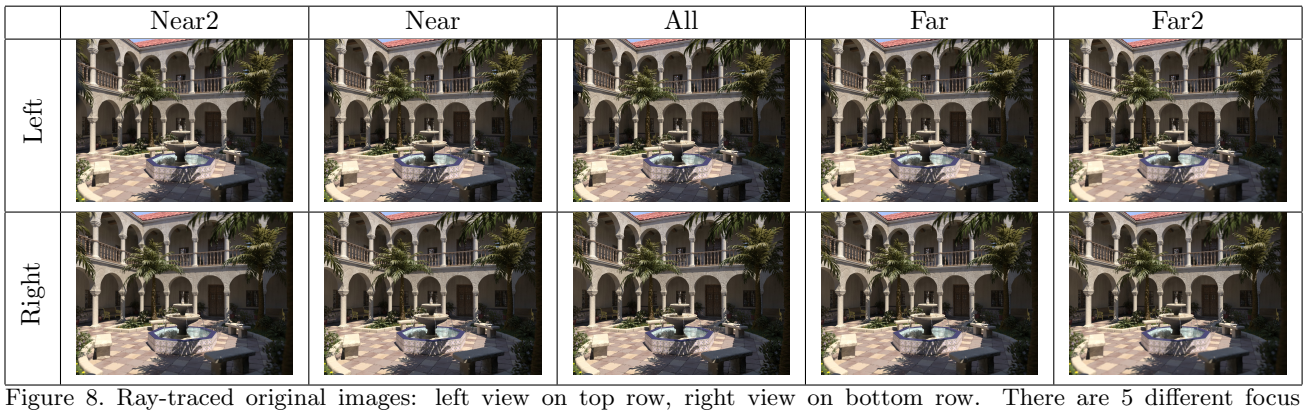

Figure 8. Ray-traced original images: left view on top row, right view on bottom row. There are 5 different focus configurations for each image: "All" has infinite DOF, "Near" and "Near2" are focused on the fountain, "Far" and "Far2" are focused on the farthest tree. "Near2" and "Far2" have a smaller DOF than "Near" and "Far".

4.2 Intermediate Results

Let us discuss the intermediate results for the (LFar,RNear) case. In Fig. 9 we can see a detail of those images: in the left image the fountain is defocused, while the flowers are focused. In the right image the fountain is focused, while the flowers are out of focus.

LFar detail RNear detail

Figure 9. Detail of the LFar and RNear original images: LFar is focused on the tree in the back, RNear is focused on the fountain.

In Fig. 10 we can see the computed disparity map, the computed Sign of the SML difference: $M(\mathbf{i})$ and the max of SML_l and SML_r, namely $w(\mathbf{i})$. The $w(\mathbf{i})$ image has been equalized in order to see details in the middle range of the values. The maximum values are very big compared to the medium - smaller ones. As expected we can observe that the $M(\mathbf{i})$ measures are very noisy. However, a sign dominance is visible for the d_{Near} and d_{Far} disparity values, corresponding respectively to the blue and green dots on the disparity map (Fig. 10a). The $w(\mathbf{i})$ image shows that reliable information can be found in textured zones, which are focused in one of both images. In Fig. 11 we can see the inputs for the Model Estimation phase described in Sec. 3.3. (a) is $M(d)$, (b) $w(d)$ and (c) $C(d)$. We can see that the minimum and maximum values of $C(d)$ are very close to the actual values of the focus distances, showing that at these disparities there is an important focus mismatch. We can easily answer the most important question: there is a focus mismatch. From the shape of the model we could also conclude that $\text{FD}_l > \text{FD}_r$ and $\text{DOF}_l = \text{DOF}_r$. However, the $\text{DOF}_l = \text{DOF}_r$ could be false if we had only access to an incomplete model: $\text{DOF}_l > \text{DOF}_r$ and $\text{DOF}_l < \text{DOF}_r$ are possible candidates, too.

Finally, in Fig. 12 we present the obtained zebras on the original images for our (LFar,RNear) example (left is focused on the tree in the back, right is focused on the fountain). We can observe that the correct regions of each image have been precisely marked.

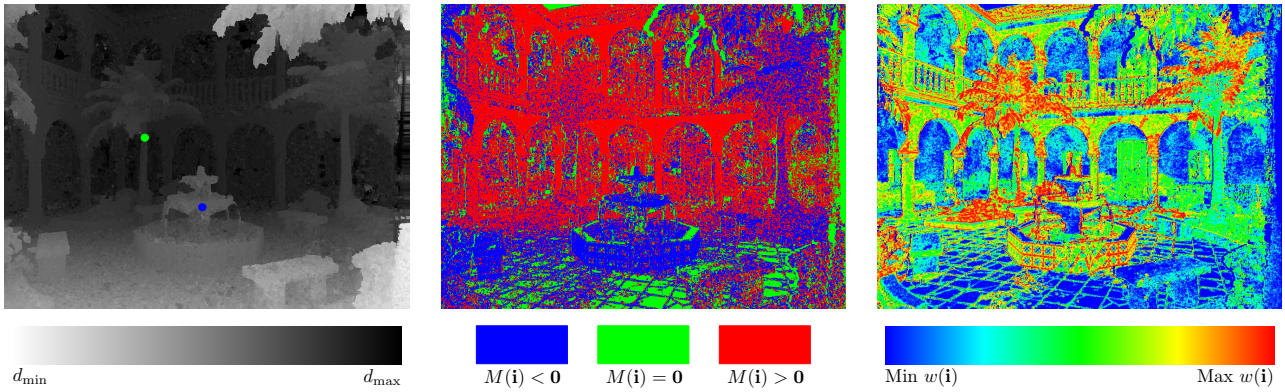

Figure 10. Intermediate results for the (LFar,RNear) test case. From left to right: (a) Disparity: $d(\mathbf{i})$, (b) Sign of SML Difference: $M(\mathbf{i})$, (c) Max of SML: $w(\mathbf{i})$. The blue and green dots in the disparity map correspond to focus distances for the Near and Far sample images.

4.3 Global Results discussion

In order to validate our approach we first tested our algorithm using the ground-truth disparity map of the scene on all possible combinations of the five images for each view. The 25 obtained results are presented in Fig. 13. We want to be able to answer the following questions, sorted by order of difficulty:

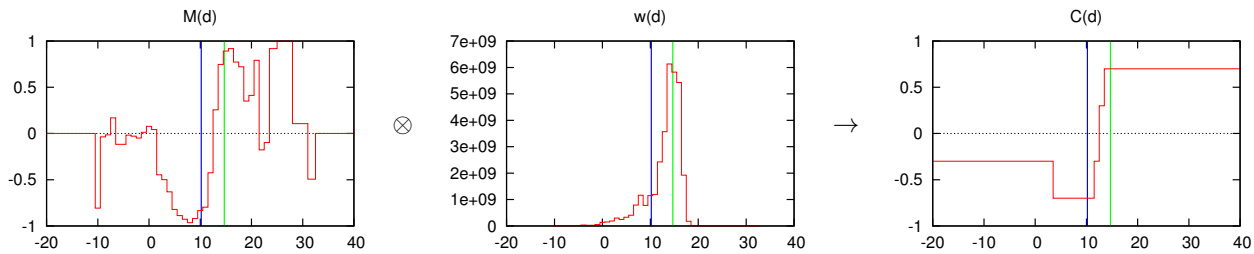

Figure 11. Results for the (LFar,RNear) test case. From left to right: (a) Weighted Mean of the sign of SML difference: $M(d)$, (b) Sum of Max of SML: $w(d)$, (c) Estimated model $C(d)$. Blue and green disparities correspond to the blue and green dots in the disparity image in Fig. 10.a.

Figure 12. The original images from the (LFar,RNear) test case with zebra stripes on the regions of the image which are less focused than in the other image: Left red zebra area (e.g. the fountain) is less focused than in the right image. Right red zebra area (e.g. the background) is less focused than in the left image.

1. Are both focus distances and depth of field perfectly matched?
2. Are both focus distances equal? Which one is bigger?
3. Are both depths-of-field equal? Which one is bigger?

In all cases we can correctly answer the first question: Diagonal elements of Fig. 13 are zero, and non-diagonal elements are not zero.

We can answer the second question accurately when the focus distances are different (i.e. near vs. far or far vs. near), as can be seen in the four lower-left corner elements and the four upper-right corner elements of Fig. 13. For the first four we get the same shape as in the example discussed in Sec. 4.2 concluding that $\mathrm{FD}_l > \mathrm{FD}_r$. For the second four group we get the y-symmetric curve, thus being able to conclude that $\mathrm{FD}_l < \mathrm{FD}_r$. In all eight cases we are unable to tell if there is a difference between both DOF as previously discussed in Sec. 4.2.

The 12 remaining cells correspond to a difference in depth-of-field only. We can see that only two of them have the expected shape, and thus give the correct answers to all questions: (LFar,RFar2) and (LFar2, RFar). In the remaining ten cases, although we cannot answer wether the DOF is different by looking at the curve, we can still answer the question "Which one is more focused?".

In Fig. 14, we present the results obtained using the disparity map computed from each image pair. We observe that the results are very similar to those obtained using the ground truth disparity. Notice that there are no significative differences between both figures.

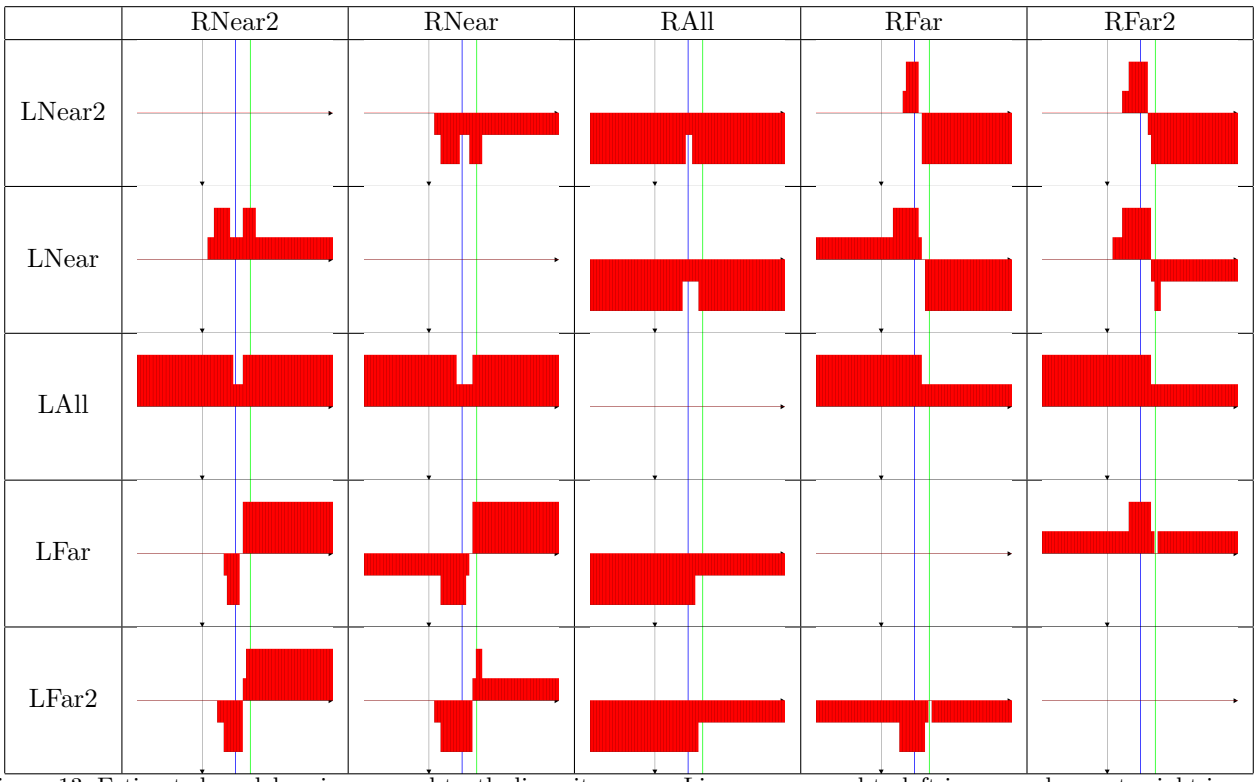

Figure 13. Estimated models using ground truth disparity maps. Lines correspond to left image, columns to right image.

Figure 14. Estimated models using computed disparity maps. Lines correspond to left image, columns to right image.

5. CONCLUSION AND FUTURE WORK

In this paper we presented a complete method for stereoscopic focus mismatch detection. Given the fact that it is impossible to recover the information lost during the shot, it is very important to have both focus distances and depth of field perfectly matched.

The results on synthetic ray-traced data are very promising. All cells in the diagonal of Fig. 14 are zero, which means that we are accurately detecting the most important case: both focus match. More than this, we do not get any false positives: all configurations having a focus mismatch are correctly identified. This means that our approach gets a 100% accuracy on the question: Are both focus distances and depth of field perfectly matched? Moreover, in the configurations where the focus are mismatched our method detects accurately the difference in focus distance. We can tell which focus is in front or behind the other. Our method does not properly detect a difference in the depth of field in all cases, however it is always at least capable of accurately telling which camera is less in focus.

The fact that we get very similar results with the ground truth disparity map and with the computed one, confirms that the chosen method for the disparity map computation gets good enough results, even with defocused images.

All the proposed algorithm steps can be performed in real-time: disparity map computation, SML Difference computation, Model fitting, model classification and giving feedback to the operator. The final method could thus be implemented to run in real time. This means that this algorithm could be integrated into onset equipment to help the operators so that their focus match perfectly before the shot.

Future work should aim at the validation of the proposed method with real footage from actual cameras. It would also be important to integrate the algorithm into a real time monitoring system to prove its performance, and validate the zebra feedback method.

Further work based on this article could be a better characterization of the image blur. Once we have detected that one image is less in focus than the other we could try to characterize the detected error. It's known, that when shooting stereoscopic footage with a mirror rig, the quality of the mirror may introduce further deformations in the reflected image. For example, if the mirror is not perfectly flat we may get anisotropic focal blur, which corresponds to optical astigmatism.

REFERENCES

[1] Lambooij, M. T. M., IJsselsteijn, W. A., and Heynderickx, I., "Visual discomfort in stereoscopic displays: a review," in [*Proc. SPIE Stereoscopic Displays and Virtual Reality Systems XIV*], **6490** (2007).

[2] Kooi, F. L. and Toet, A., "Visual comfort of binocular and 3D displays," *Displays* **25**, 99–108 (Aug. 2004).

[3] Seuntiens, P., Meesters, L., and Ijsselsteijn, W., "Perceived quality of compressed stereoscopic images: Effects of symmetric and asymmetric JPEG coding and camera separation," *ACM Trans. Appl. Percept.* **3**, 95–109 (Apr. 2009).

[4] Stelmach, L., Tam, W., Meegan, D., Vincent, A., and Corriveau, P., "Human perception of mismatched stereoscopic 3D inputs," in [*International Conference on Image Processing (ICIP)*], **1**, 5–8 (2000).

[5] Kroon, R. W., "3D A to Z," in [*3D TV and 3D Cinema*], Mendiburu, B., ed., 195 – 232, Focal Press, Boston (2012).

[6] FitzGerrell, A. R., Edward R. Dowski, j., and Cathey, W. T., "Defocus transfer function for circularly symmetric pupils," *Appl. Opt.* **36**, 5796–5804 (Aug. 1997).

[7] Subbarao, M. and Surya, G., "Depth from defocus: A spatial domain approach," *International Journal of Computer Vision* **13**, 271–294 (1994).

[8] Mennucci, A. and Soatto, S., "On observing shape from defocused images," in [*Image Analysis and Processing, 1999. Proceedings. International Conference on*], 550 –555 (1999).

[9] Pentland, A. P., "A new sense for depth of field," *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)* **9**, 523 –531 (July 1987).

[10] Subbarao, M., "Parallel depth recovery by changing camera parameters," in [*Intl. Conf. on Computer Vision.*], 149 –155 (Dec. 1988).

[11] Lin, H.-Y. and Gu, K.-D., "Photo-realistic depth-of-field effects synthesis based on real camera parameters," in [*Advances in Visual Computing (ISVC 2007)*], *Lecture Notes in Computer Science* **4841**, 298–309, Springer-Verlag (Nov. 2007).

[12] Rajagopalan, A., Chaudhuri, S., and Mudenagudi, U., "Depth estimation and image restoration using defocused stereo pairs," *IEEE Trans. on Pattern Analysis and Machine Intelligence* **26**, 1521 –1525 (Nov. 2004).

[13] Schechner, Y. and Kiryati, N., "Depth from defocus vs. stereo: how different really are they?," in [*Proc. Intl. Conf. on Pattern Recognition*], **2**, 1784–1786 (Aug. 1998).

[14] Sizintsev, M. and Wildes, R. P., "Coarse-to-fine stereo vision with accurate 3D boundaries," *Image and Vision Computing* **28**(3), 352 – 366 (2010).

[15] Huang, W. and Jing, Z., "Evaluation of focus measures in multi-focus image fusion," *Pattern Recognition Letters* **28**(4), 493–500 (2007).

[16] Nayar, S. and Nakagawa, Y., "Shape from focus," *IEEE Trans. on Pattern Analysis and Machine Intelligence* **16**, 824–831 (Aug. 1994).