Can Stereo Vision replace a Laser Rangefinder?

M. Antunes, J.P. Barreto, C. Premebida and U. Nunes

Abstract— **Many robotic systems combine cameras with Laser Rangefinders (LRF) for simultaneously achieving multipurpose visual sensing and accurate depth recovery. Employing a single sensor modality for accomplishing both goals is an appealing proposition because it enables substantial savings in equipment, and tends to decrease the overall complexity of the system. This article explores the possibility of replacing LRF by passive stereo vision for reconstructing the scene along a 2D scan plane. We present a new stereo algorithm that is specifically tailored for the purpose. The algorithm recovers the depth along the scan plane using a symmetry-based matching cost (SymStereo), and refines the raw estimates by applying dynamic programming, followed by a Markov Random Field (MRF) that decides if the reconstructed contour is a line or not. We report for the first time comparative experiments between Stereo Rangefinding (SRF) and LRF. The results are encouraging by showing that SRF can be a plausible alternative to LRF in several application scenarios. Moreover, since SRF also enables independent depth estimates along multiple scan planes with arbitrary orientation, being the only constraint that the scan plane intersects the stereo baseline, it is an important benefit that can be decisive for many robotic applications.**

I. Introduction

There are many applications in robotics that make simultaneous use of visual data and laser-scans. A non-exhaustive list of examples includes autonomous navigation [1], pedestrian detection [2], object classification [3], and self-localization in indoor and outdoor scenes [4]. Laser Rangefinding is popular because it enables accurate depth measures in real-time, being effective under most operating conditions like long range, no illumination, lack of surface textures, etc. On the other hand, passive vision is an extremely versatile sensor modality, providing rich image information that can be used for multiple purposes ranging from obstacle detection to object recognition, passing by texture classification, self-localization and mapping, and human-machine interfaces[1].

Replacing two sensor modalities by a single one without sacrificing skills or system capabilities is an appealing proposition. This would lead to savings in equipment with a positive impact in the overall cost of the final system. Moreover, it would contribute to decrease the R&D effort, namely by avoiding complex middleware for the synchronous acquisition of different sensor modalities. Other possible advantages are payload reduction, which is specially important in flying robots and aero-spatial systems [5], and enhanced autonomy by increasing the battery-life.

It is unlikely that LRF can ever replace passive vision without losses in versatility and system capabilities. Fortunately, the opposite seems much more feasible specially in cases where two calibrated cameras are available. It is well known that stereo enables 3D reconstruction by densely associating pixels across images and triangulating. Thus, it is plausible that it can succeed in estimating depth along a scan plane with an accuracy close to LRF. Please note that for the case of robots that originally combine monocular vision with laser, the replacement of a LRF by a second camera is still an attractive proposition: the camera is in general less expensive, the synchronization of image acquisition is a trivial matter, and both the payload and energy consumption tend to decrease.

This work is motivated by the possibility of replacing LRF by stereo vision in robotic applications. The article proposes a new stereo algorithm that is specifically tailored for estimating depth along an arbitrary virtual scan plane. Since the resulting profile cut of the scene resembles the one that would be obtained by LRF, the technique is named *Stereo Rangefinding* (SRF). The depth estimates obtained with SRF are compared against real range data acquired by a state-of-the-art LRF equipment. The experiment clearly shows the strengths and weaknesses of each technology, providing evidence that passive stereo can be an alternative to LRF in certain application contexts.

A. Contributions

Although dense stereo is a well studied topic [6], several aspects must be addressed in order for stereo to be an alternative to LRF: First, the objective of most stereo methods is to assign discrete disparity labels to image pixels, which does not necessarily mean to accurately estimate metric depth. As shown latter in the paper, perfect image matching results often correspond to reconstructions that are inaccurate when compared to LRF measurements. Second, while LRF recovers depth along a scan plane, stereo provides depth estimates for every pixel in the image. A possible solution for mimicking the LRF output is to fully reconstruct the scene and intersect the result with the scan plane. This is highly inefficient because it requires unnecessary computation that will difficult real-time implementations. Third, stereo matching is often ambiguous in low textured surfaces, lacking the robustness of LRF to different operating scenarios.

This paper proposes a new stereo algorithm for addressing the difficulties mentioned above. While most stereo methods rely in photo-consistency for matching cost, we use induced

The authors are with the Department of Electrical and Computer Engineering, Institute of Systems and Robotics, University of Coimbra, Portugal. {michel, jpbar, cpremebida, urbano}@isr.uc.pt

[1] Note that replacing a LRF and a camera by a RGB-D camera is not a viable alternative due to the large depth ranges of operation and variable lighting conditions.

TABLE I
SPECIFICATIONS OF THE CAMERA AND THE LRF

Camera		LRF	
Manufacturer	Point-Grey	Manufacturer	Sick
Baseline	≈ 45 cm	Model	LMS200
Resolution	1280×960	Horiz. Res.	0.25^o

image symmetry for obtaining an energy that quantifies likelihood of pixel correspondence. This framework, called SymStereo, has been recently introduced in [7] where it is proved to be highly effective in estimating depth along virtual scan planes passing between the cameras. According to [8], SymStereo is the top performer metric for obtaining profile cuts of the scene, being specially resilient to situations of low texture.

We also propose several strategies for refining the energy results generated by SymStereo, and accurately locate the profile cut $\mathcal{C}$ where the virtual scan plane meets the scene. Since $\mathcal{C}$ is usually the union of a limited number of continuous contours, we use dynamic programming (DP) for enforcing connectivity between point detections across contiguous epipolar lines. In addition, it is also assumed that the contours in $\mathcal{C}$ are often line segments corresponding to plane surfaces that dominate man-made environments. Thus, and in parallel to DP, we run a weighted Hough transform for extracting lines in the energy output of SymStereo. This procedure gives raise to more than one hypothesis for the intersections between epipolar lines and the profile cut $\mathcal{C}$. The decision problem is formulated as a Markov Random Field (MRF) that assigns to each epipolar line a unique label. This label either corresponds to the DP result, or to one of the lines extracted with the Hough transform.

The approach for SFR is experimentally evaluated using a setup specially built for the purpose (see Fig. 1). The setup combines two cameras with a 2D LRF that are carefully calibrated for enabling direct comparison between the metric results achieved with stereo and laser. To the best of our knowledge this is the first experiment of the kind reported in the literature. In [9], Morales and Klette evaluate dense disparity maps against the 3D range data obtained with LRF. However, their objective is to obtain reliable ground truth for benchmarking different stereo algorithms, rather than comparing the accuracy of the two sensor modalities. Hrabar presents in [10] a comparison of stereo vision and laser-based range sensing for rotorcraft unnamed aerial vehicle obstacle avoidance. In this work the main evaluation criteria is not the metric depth, but instead the success rate in obstacle avoidance.

II. EXPERIMENTAL SETUP

We briefly introduce the experimental setup for the synchronous acquisition of stereo images and range data. The setup combines a 2D LRF with two perspective cameras for which the specifications are provided in Tab.I. The sensors are mounted on a rigid mobile platform with the laser placed between the cameras as shown in Fig. 1. The camera baseline is around 45 cm and the distance between the top camera and the laser is roughly 19.5 cm. The cameras are not aligned, and $\mathbf{C}_1$ is assumed to be the reference view.

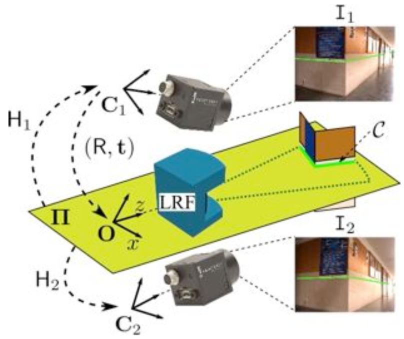


Fig. 1. Experimental setup. The top camera $\mathbf{C}_1$ (reference view) points down, the bottom camera $\mathbf{C}_2$ points up, and the LRF is positioned between the cameras. The LRF recovers the profile cut $\mathcal{C}$ that is the contour where the scan plane $\mathbf{\Pi}$ meets the scene structure. $\mathcal{C}$ is projected in the two images $\mathtt{I}_1$ and $\mathtt{I}_2$ using the extrinsic calibration results.

Referring to Fig. 1, the stereo cameras are calibrated using Bouguet's calibration toolbox [11], and the relative pose between the LRF and the reference camera $\mathbf{C}_1$ is estimated using the minimal solution proposed by Vasconcelos et al. [12]. This enables to determine the homogeneous representation of the scan plane $\mathbf{\Pi}$ in the stereo coordinate system, and compute the plane-homographies H_1 and H_2 that accurately map range data into images $\mathtt{I}_1$ and $\mathtt{I}_2$.

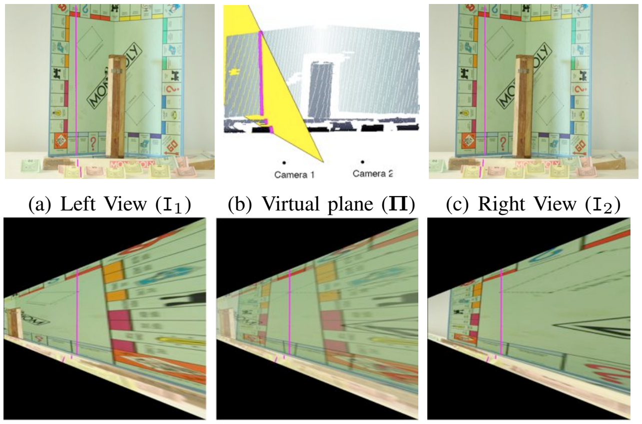
III. STEREO RANGEFINDING

This section describes the SymStereo framework that is used to generate a 2D energy signal E given the stereo images $\mathtt{I}_1$ and $\mathtt{I}_2$, and the plane-homographies H_1 and H_2 that relate the scan plane $\mathbf{\Pi}$ with each view. The energy E can be understood as an indirect measurement of the likelihood of a point location in $\mathbf{\Pi}$ to lie in the profile cut $\mathcal{C}$, with $\mathcal{C}$ being the contour where the scan plane intersects the scene surfaces.

A. Overview of SymStereo

Plane sweeping is a well known stereo technique that was introduced in [13] for finding matches across multiple images. The basic idea consists in sampling the 3D space by a family of parallel virtual planes, back-project the images onto these planes, and find the locations where the back-projections are similar. Ideally, these locations correspond to the intersection points of the plane with the imaged surfaces, which enables depth recovery. SymStereo relates with Plane Sweeping in the sense that it also considers virtual planes in the 3D space, however there are two major differences: (i) we exclusively consider virtual planes that intersect the stereo baseline in a point between the cameras; and (ii) instead of looking for regions where back-projections correlate, the profile of intersection between the virtual plane and the scene structure is identified as a mirroring contour between back-projected images.

Most works in the literature consider virtual planes intersecting the baseline a degenerate configuration. Under



(a) Left View ($\mathtt{I}_1$) (b) Virtual plane ($\mathbf{\Pi}$) (c) Right View ($\mathtt{I}_2$)

(d) Left back. $\mathtt{I}_1^b$ (e) $\mathtt{I}_s = \mathtt{I}_1^b + \mathtt{I}_2^b$ (f) Right back. $\mathtt{I}_2^b$

Fig. 2. The Monopoly sequence [6] is used to illustrate our approach (note that it corresponds to the traditional stereo setup, in which the images are aligned side-by-side.). The virtual scan plane $\mathbf{\Pi}$ passes between the cameras and intersects the scene in a discontinuous curve $\mathcal{C}$ (b), called the *profile cut*. (a) and (c) show the two input views, $\mathtt{I}_1$ and $\mathtt{I}_2$, with the projection of C overlaid. $\mathtt{I}_1^b$ and $\mathtt{I}_2^b$ are obtained by back-projecting $\mathtt{I}_1$ and $\mathtt{I}_2$ on $\mathbf{\Pi}$. The two back-projections are mirrored with respect to the C (d) and (f). Adding $\mathtt{I}_1^b$ and $\mathtt{I}_2^b$ results in an image $\mathtt{I}_s$ that is symmetric around C (e), while subtracting gives rise to a anti-symmetric signal.

such circumstances the left and right back-projections do not overlap in the locations where the plane meets the structure, but become reflected one with respect to the other (see Fig. 2). Thus, the sum of both back-projections gives rise to an image signal that is locally symmetric around the profile cut, while the subtraction results in a signal that is anti-symmetric. These symmetries are usually not "strict symmetries" due to perspective distortion, surface slant and occlusions, but can be used as cues to recover the profile cut $\mathcal{C}$ where the virtual plane meets the scene.

As input SymStereo receives a pair of calibrated stereo views, $\mathtt{I}_1$ and $\mathtt{I}_2$, and a pre-defined virtual plane that in the particular case of our application is the laser scan plane $\mathbf{\Pi}$. The orientation of the virtual plane is arbitrary with the only requirement being that it intersects the baseline without going through the camera centers (Fig. 3(a)). The two stereo images are back-projected onto $\mathbf{\Pi}$ using the homography mappings, and the warping results are added and subtracted giving rise to signals $\mathtt{I}_s$ and $\mathtt{I}_a$. The former is locally symmetric around the contour $\mathcal{C}$ that we aim to estimate, while the latter is anti-symmetric.

The back-projection onto the virtual scan plane $\mathbf{\Pi}$ is not trivial and Sec. III-B describes in detail how the rendering of symmetric and anti-symmetric image signals can be accomplished in a computationally efficient manner. The warping results $\mathtt{I}_s$ and $\mathtt{I}_a$ are properly rectified with the epipolar lines being aligned. Under such circumstances, the contour $\mathcal{C}$ intersects each epipolar line of the image signals $\mathtt{I}_s$ and $\mathtt{I}_a$ in a single point, and the symmetry analysis can be independently carried for each line using a 1D log-Gabor transform [14]. This process is described in Sec. III-C, with $\mathtt{I}_s$ and $\mathtt{I}_a$ giving raise to the symmetry and anti-symmetry energy images $\mathtt{E}_s$ and $\mathtt{E}_a$, that are multiplied to generate the final energy E.

B. Generation of symmetric/antisymmetric signals

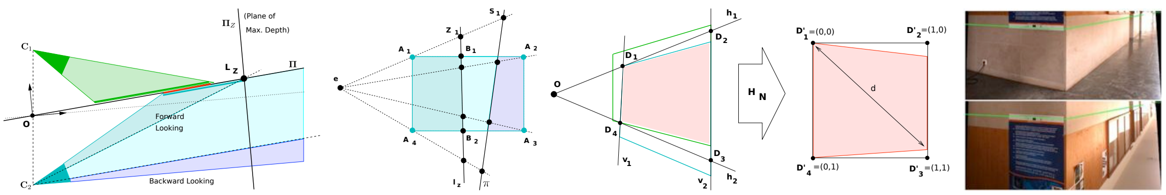
Consider a virtual scan plane $\mathbf{\Pi}$, with an arbitrary normal orientation, that intersects the baseline in the point $\mathbf{O}$ (see Fig. 3). As discussed above, by back-projecting the stereo views onto $\mathbf{\Pi}$, it is possible to generate two distinct image signals $\mathtt{I}_s$ and $\mathtt{I}_a$ that are symmetric and anti-symmetric at the locations where the plane cuts the scene structure. This section shows how to render the back-projection images $\mathtt{I}_1^b$ and $\mathtt{I}_2^b$ efficiently.

Without lack of generality we assume a maximum value for the scene depth (note that the working range of the LRF is set to $80m$, so that for the experiments we also set this maximum depth for SymStereo), which means that the profile cut $\mathcal{C}$ must lie on the area spreading between $\mathbf{O}$ and line $\mathbf{L}_z$ where the plane of maximum depth meets $\mathbf{\Pi}$ (Fig. 3(a)). Thus, for each image $\mathtt{I}_1$ and $\mathtt{I}_2$ we can define an interest region by the following steps (refer to Fig. 3(b)):

1) Determine lines $\mathbf{l}_z$ and π by projecting $\mathbf{L}_Z$ and the line at infinity using H_1 and H_2
2) For each image corner $\mathbf{A}_i$, consider the line defined by the corner and the epipole $\mathbf{e}$, and determine the intersections $\mathbf{Z}_i$ and $\mathbf{S}_i$ with $\mathbf{l}_z$ and π
3) If the cross-ratio $\{\mathbf{Z}_i, \mathbf{e}; \mathbf{A}_i, \mathbf{S}_i\}$ is negative, then the corner $\mathbf{A}_i$ is in the interest region, otherwise it is outside; if the cross-ratios are all positive then the interest region is empty, if the cross-ratios are all negative then the interest region is the entire image, otherwise the interest region is the polygon defined by the corners $\mathbf{A}_i$ with negative cross-ratio and the intersections of $\mathbf{l}_z$ with the image borders.

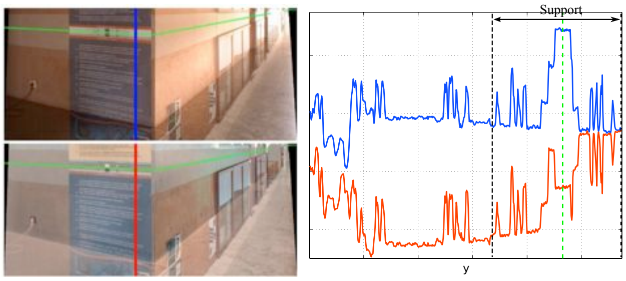
The profile cut C can only be recovered if it is simultaneously seen in both views. Thus, the search region can be further constrained by back-projecting the boundaries of the bottom and top interest regions onto $\mathbf{\Pi}$ and finding their intersection polygon (Fig. 3(c)). Mapping the polygon back into the input views yields the image regions that must be warped. Signals $\mathtt{I}_1^b$ and $\mathtt{I}_2^b$ of Fig. 2 were obtained by warping the image search regions by the inverse of H_1 and H_2 using an uniform plane tessellation.

Two issues remain: (i) the epipolar lines are not vertically aligned, which can complicate subsequent processing and (ii) a uniform plane tessellation does account for the original image resolution, causing a magnification that increases with depth. We address these problems by rectifying the back-projections using a normalizing transformation H_N. H_N is a projective transformation on the cut plane $\mathbf{\Pi}$ that inscribes the search polygon in an unitary square as shown by Fig. 3(c). Lines $\mathbf{h}_1$ and $\mathbf{h}_2$, that join the origin $\mathbf{O}$ with the top and bottom vertex of the polygon, are mapped into the top and bottom sides of the square. This grants that epipolar lines become vertically aligned. Lines $\mathbf{v}_1$ and $\mathbf{v}_2$ are chosen so that the transformed polygon is enclosed by the square and has maximum area. The resolution of the tessellation is determined by averaging the pixel length of the diagonal d that is mapped back in the two stereo images. The



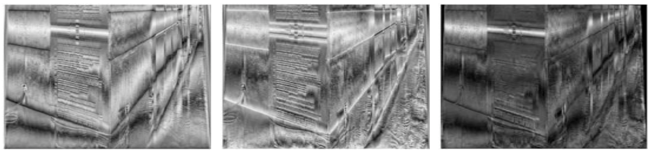
(a) Side View (b) Image of $\mathbf{C}_2$ (c) Normalization using H_N (d) $\mathtt{I}_1^b$ and $\mathtt{I}_2^b$

Fig. 3. Back-projection onto a virtual scan plane $\mathbf{\Pi}$ with arbitrary normal orientation. (a) side view where the scene structure is assumed to lie between the cameras and the plane of maximum depth $\mathbf{\Pi}_Z$. The entire top image is considered an interest region because it back-projects in the area between the origin $\mathbf{O}$ and the line $\mathbf{L}_Z$. In the case of view $\mathbf{C}_2$ the bottom image side is back-projected behind $\mathbf{O}$, while the middle part of the image is back-projected beyond $\mathbf{L}_Z$. Thus, the interest region is limited to the top most side (b). The search for the contour where $\mathbf{\Pi}$ cuts the scene surfaces needs only to be carried in the polygon of intersection of the left and right interest regions (red). (c) The alignment of the epipolar lines and the definition of a suitable tessellation are achieved by inscribing the search polygon into a unit square using the homography H_N. (d) shows the results of warping the bottom and top images of Fig.1 according to $\mathsf{H}_1^{-1}\mathsf{H}_N$ and $\mathsf{H}_2^{-1}\mathsf{H}_N$.



(a) $\mathtt{I}_s$(top) and $\mathtt{I}_a$(bottom) (b) 1-D Signal (Epipolar Line)

Fig. 4. (a) The image signals $\mathtt{I}_s$ and $\mathtt{I}_a$. The profile cut $\mathcal{C}$ is shown in green, and it is rather clear that $\mathtt{I}_s$ and $\mathtt{I}_a$ have a vertical symmetry and anti-symmetry in the points of the contour. (b) shows the intensity level of $\mathtt{I}_s$ (blue) and $\mathtt{I}_a$ (red) for one epipolar line. The position of $\mathcal{C}$ can be identified by searching for pixel locations that present the largest support for both a local symmetry in $\mathtt{I}_s$ (top) and a local anti-symmetry in $\mathtt{I}_a$ (bottom).



(a) E_s (b) E_a (c) E

Fig. 5. Energy obtained from signal symmetry analysis over Fig. 4(a) (whiter corresponds to higher energy). The energy of symmetry E_s has several peaks, many of which are in locations other than the contour we aim to detect. This is typically due to natural symmetric structures appearing in $\mathtt{I}_s$. A similar observation can be made for the case of the anti-symmetry. However, these natural symmetries and anti-symmetries rarely occur at the same location in $\mathtt{I}_s$ and $\mathtt{I}_a$. The pixel-wise multiplication of E_s and E_a yields E, in which many spurious detections are filtered out and the profile cut becomes salient.

results for warping the stereo views shown in Fig.1 by the transformations $\mathsf{H}_1^{-1}\mathsf{H}_N$ and $\mathsf{H}_2^{-1}\mathsf{H}_N$ are shown in Fig. 3(d).

The symmetric and anti-symmetric images $\mathtt{I}_s$ and $\mathtt{I}_a$, that will serve as input to the next stage, are obtained by computing the sum and difference of $\mathtt{I}_1^b$ and $\mathtt{I}_2^b$. Fig. 4 shows the generated signals for the example that we are using for illustrating the approach. $\mathtt{I}_s$ is clearly symmetric in the points of the profile contour $\mathcal{C}$ along a particular epipolar line, while $\mathtt{I}_a$ is clearly anti-symmetric. Fig. 4(b) considers the image intensities in more detail. The local signal symmetries and anti-symmetries at the profile cut location are not perfect because of the joint effects of noise, sampling, perspectivity, surface slant and occlusions. Nevertheless, by searching for the location that has the largest symmetry support in the top image and anti-symmetry in the bottom image, we can clearly identify the correct profile cut point.

C. Measuring image symmetry

Each cut plane $\mathbf{\Pi}$ gives rise to a pair of image signals $\mathtt{I}_s$ and $\mathtt{I}_a$ that are respectively symmetric and anti-symmetric at the profile cut $\mathcal{C}$. This section briefly discusses the quantification of local signal symmetries aimed at locating $\mathcal{C}$. Since $\mathtt{I}_s$ and $\mathtt{I}_a$ are rectified with respect to the epipolar geometry, the profile contour intersects each image column in no more than one point, and the symmetry has vertical orientation. Therefore, we will carry the symmetry detection in 1-D independently for each image column.

Our objective is to quantify local symmetry along points in a 1D signal. Moreover, the chosen algorithm must handle low textured regions and be computationally efficient. Taking these specifications into account, we adapted for the 1D case the approach proposed by Kovesi [14] that uses a log-Gabor wavelet transform for measuring symmetry at pixel image locations. Kovesi shows that symmetry points give rise to specific phase patterns in the Fourier series of the signal [14]. Thus, he proposes to measure local symmetry and anti-symmetry by applying a bank of log-Gabor filters at different wavelet scales. The output is the energy of symmetry along the columns of image $\mathtt{I}_s$, and the energy of anti-symmetry along the columns of image $\mathtt{I}_a$ (Fig. 5(a) and (b)). Since the profile cut $\mathcal{C}$ is at the same image location in $\mathtt{I}_s$ and $\mathtt{I}_a$, the two energy signals are multiplied yielding the joint energy signal E (Fig. 5(c)). As shown in Fig. 5(c), in textured regions the image of the profile cut can be clearly identified due to a strong energy ridge. However, in low-textured regions the energy is widely dispersed around the correct location. We will see in the next section that it is possible to accurately estimate the profile cut using global approaches, and implicitly recover depth along the virtual scan plane.

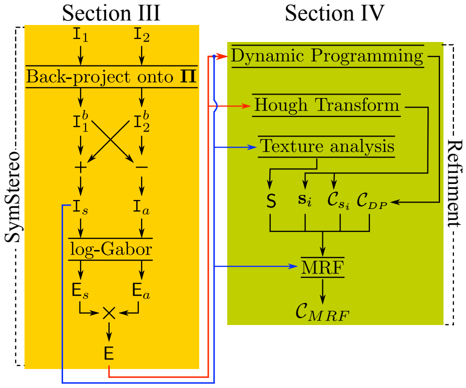


Fig. 6. Overview of the SRF framework. The SymStereo algorithm is used for calculating the symmetry image $\mathtt{I}_s$ and the joint energy E, which are used as input for the refinement steps.

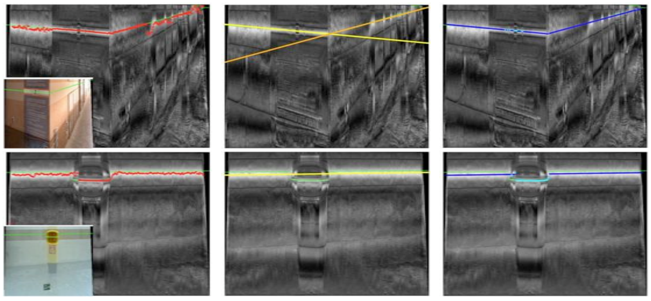

(a) $\mathcal{C}_{DP}$ (b) $\mathcal{C}_{s_i}$ (c) $\mathcal{C}_{MRF}$

Fig. 7. 2 examples for the estimation of the profile cut location, green is the range data of the LRF. (a) estimated using Dynamic Programming (red), the correct location is only detected in textured regions or near strong edges; (b) lines (yellow and orange) detected using the hough transform, corresponding to the intersection of the virtual scan plane with planar scenes; and (c) MRF labeling decides between $\mathcal{C}_{DP}$ (cyan) and $\mathcal{C}_{s_i}$ (blue).

IV. Refinement for SymStereo

As described previously, the contour $\mathcal{C}$ that we aim to estimate intersects each column of the image $\mathtt{I}_1^b$ in a single point. A naive approach for locating $\mathcal{C}$ is to simply select the energy maxima in E along each column. However, and as shown in Fig. 5, this would lead to very noisy estimates. This section proposes to improve the estimation of the profile cut by considering two soft constraints in parallel:

1) The point locations where the profile cut intersects contiguous epipolar lines are likely to be connected. This prior can be enforced using dynamic programming (DP) for detecting the contour $\mathcal{C}$ in the energy image E (Sec. IV-A).
2) For the case of operation in man-made environments, the segments in the profile cut $\mathcal{C}$ are often line segments that correspond to the intersection of the scan plane with planar surfaces in the scene. This prior can be enforced by using the Hough Transform weighted by the energy E (Sec. IV-B).

The DP provides a generic contour estimate that is not a line, while the Hough Transform provides N line estimates. The decision about the correct hypothesis for the location of the profile cut in each image column of $\mathtt{I}_1^b$ is taken by a Markov Random Field (MRF) formulation (Sec. IV-C). The overall flow of the algorithm is depicted in Fig. 6, in which the input to the refinement steps is the symmetry image $\mathtt{I}_s$ and the joint energy E.

A. Dynamic Programming (DP)

We use a simple optimization approach for obtaining a binary labeling of the image $\mathtt{I}_1^b$, so that each image column x has only one pixel y set to one, that corresponds to the most likely pixel location lying in the profile cut. This is accomplished using a straightforward DP approach, similar to the scanline optimization that is described in [6]. The algorithm works by computing the maximum cost path in E, where the energy for each pixel $\mathbf{p}$=(x_p, y_p) is re-defined as:

$$\mathsf{D}(x_p, y_p) = \mathsf{E}(x_p, y_p) + \min_y \left(\mathsf{D}(x_p-1, y) + V_{DP}(y_p, y)\right) \quad (1)$$

with V_{DP} being a smoothness term given by

$$V_{DP}(y_p,y) = \begin{cases} \frac{\lambda_{DP}}{|I_s(x_p,y_p)-I_s(x_p-1,y)|} & \text{if } |y_p - y| > 0 \\ 0 & \text{otherwise} \end{cases}$$

where λ_{DP} is a constant parameter. The binary labeling is accomplished by selecting for each image column x_p the pixel (x_p, y_p) with maximum cost D. Thus, we obtain a contour corresponding to a possible location of the profile cut expressed in discrete terms, according to the plane tessellation described in Sec. III-B. In order to refine the contour estimation and obtain sub-pixel precision, we fit to each x_p a parabola in E around the neighborhood of y_p. The output of this step is the profile cut location $\mathcal{C}_{DP}$. Fig.7(a) shows the estimation of the contour $\mathcal{C}_{DP}$ detected in two stereo examples. As can be seen, $\mathcal{C}_{DP}$ is only accurate in textured regions and near strong intensity edges.

B. Line detection using the Hough transform

In order to detect straight lines in $\mathtt{I}_1^b$, which correspond to the intersection of $\mathbf{\Pi}$ with a planar scene surface, a weighted Hough Transform is applied to the joint energy E. We extract at most N line segments $\mathcal{C}_{s_i}$ with parameters $\mathbf{s}_i$, where $i = 1, ..., N$ (a threshold for the minimum length of each line segment is used). Fig.7(b) shows the contours $\mathcal{C}_{s_i}$ estimated using the Hough Transform. We are able to accurately locate the profile cut segments that are in accordance to the assumed prior even in very low and repetitive textured regions.

C. MRF for straight and non-straight profile cut labeling

Given a particular epipolar line, there are $N + 1$ possible locations for the profile cut, N corresponding to the extracted lines segments $\mathcal{C}_{s_i}$ and one corresponding to the estimation $\mathcal{C}_{DP}$ using dynamic programming. In order to decide which one is the most suitable point on each epipolar line, we formulate the decision as a labeling problem in a Markov Random Field (MRF). The objective is to assign to each image column x in the set X a label l_x in the set $\mathcal{L}$, which is the union of all straight line segments ($\mathcal{C}_{s_i}$) and the non-straight label corresponding to $\mathcal{C}_{DP}$. The energy to minimize is given by:

$$E = \sum_{x \in X} d(l_x) + \lambda_{MRF} \sum_{k \in \mathcal{N}_x} V_{xk}(l_x, l_k),$$

where $\mathcal{N}_x$ is the 2-neighborhood defined by the two adjacent columns k of x, λ_{MRF} is a constant parameter, and l_x is either assigned to one of the labels s_i (straight line) or DP (general contour). The data function is defined as:

$$d(l_x) = \begin{cases} -(\mathsf{E}_{l_x}(x) + \gamma_S(1 - \mathsf{S}(x, l_x))) & \text{if } l_x = DP \\ -\mathsf{E}_{l_x}(x) & \text{otherwise} \end{cases}$$

where $E_{l_x}(x) = E(x, C_{l_x}(x))$, S denotes the image entropy in the neighborhood of x for measuring the texture, and γ_S is a constant parameter. We use S for penalizing the label DP in low-textured regions. Finally, the smoothness term is given by:

$$V_{xk}(l_x, l_k) = \begin{cases} 0 & \text{if } l_x = l_k \\ |\mathcal{C}_{l_x}(x) - \mathcal{C}_{l_k}(k)| & \text{if } (l_x \vee l_k) = DP \\ \min(|\mathcal{C}_{l_x}(x) - m_{l_x l_k}|, |\mathcal{C}_{l_k}(k) - m_{l_x l_k}) & \text{if } (l_x \wedge l_k) = straight \end{cases}$$

where $m_{l_x l_k}$ is the intersection point between lines $\mathbf{s}_x$ and $\mathbf{s}_k$, which aims to penalize transitions between line contours that are far away from the corresponding point of intersection. The energy is minimized using graph-cuts [15], and the final estimation for the profile cut C_{MRF} is directly obtained from the MRF labeling. Fig.7(c) shows the estimation of the contour $\mathcal{C}_{MRF}$. As can be observed, the MRF correctly distinguishes between straight and non-straight segments of the contour $\mathcal{C}$, as well as decides correctly the most suitable point of transition between the line contributions.

V. Experimental results

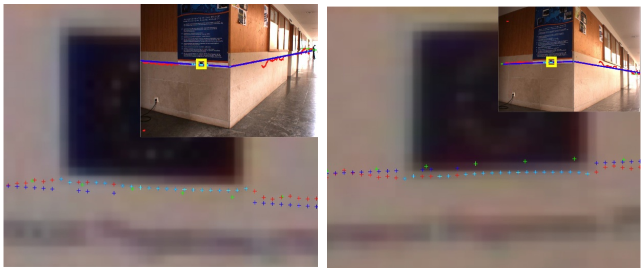
This section compares the depth estimates achieved with our algorithm for SRF[2] against real range data obtained with LRF. Fig. 8 shows pairs of stereo images and corresponding top views of the scan plane with different depth estimates overlaid. The green contour refers to the laser readings, the red points concern the depth estimates obtained by SymStereo with DP refinement, and the blue contour represents the final results after MRF labeling, with dark blue denoting straight line segments (the Hough Transform estimates), and light blue denoting non-straight segments (the DP estimates). The different profile cuts are projected onto the stereo views for analysis purposes. The examples try to cover a broad range of operating conditions including indoor and outdoor scenes, planar and non-planar surfaces, variable illumination, low textured regions, and high surface slant. The input data consists in real image pairs that are challenging for most dense stereo methods. The objective is to evaluate till which extent we succeeded in leveraging the performance of passive stereo to reach the metric accuracy observed for LRF.

The overall results are quite encouraging. For many cases the profile cut estimated using the SRF algorithm of Fig. 6 is coincident, or a few millimeters away, of the range data. SymStereo followed by DP provides accurate depth estimates whenever the profile cut $\mathcal{C}$ lies in textured surfaces or is close to strong edges. This can be clearly observed in the bottom of the poster of cases (a) and (b), the vertical wood support of case (g), the yellow container of case (i), and the pedestrian of case (j). Texture and edges are strong image cues that are readily captured by SymStereo. On the other hand, the DP depth results are often inaccurate in low-textured regions because the symmetry energy E tends to become disperse around the contour location, and the path optimization is unable to handle the ambiguity. This can be observed in almost every wall in the examples of Fig. 8, with the metric depth estimates being noisy and spread around the laser range data. Fortunately, and for the case of planar surfaces, the line segment prior followed by MRF selection seems to be effective in correcting most of the errors. This is shown in examples (d) and (g), where the line cuts are correctly determined despite of low texture, as well as in examples (a), (b), (e), (f), and (h), where the low-texture is associated with strong surface slant and high depth ranges. Also notice that the MRF correctly decides between DP and line cut estimates in the examples (g), (i) and (j).

Major failure occurs in the most distant walls of examples (b), (d), (i), and (j). Curiously, these poor estimates do not happen in the cases (e), (g), and (h), despite of similar circumstances in terms of texture, slant, and depth range. This apparent contradiction can be explained by the fact that the induced symmetry, that is quantified by SymStereo, is only perfect for a particular combination of surface slant and point $\mathbf{b_\Pi}$ where the virtual scan plane intersects the baseline (for further details see [16]). Whenever the orientation of the surface to be reconstructed differs from the surface slant that grants perfect induced symmetries for a particular $\mathbf{b_\Pi}$, then the symmetry deviation is source of errors. This problem is usually handled by the log-Gabor wavelets with wider spatial support, that use global image texture information for finding the right location of the profile cut. However, in the absence of large textured support, the symmetry deviation is not compensated by the log-Gabor wavelets, and the energy E does not present a well-defined ridge along contour $\mathcal{C}$ (see the last row of Fig.7). This is the main reason for the failures observed in examples (d), (i) and (j) of Fig.8. In the case (e) the scan plane intersects the surface further away from the white wall, which is enough for creating a wider textured support region. In the examples (g) and (h), the cameras are closer to the slanted surfaces and the texture is better perceived. Remark that in case (b), the bad depth estimation of the left wall is mainly due to the poor illumination.

So far the comparison was carried in the scan plane considering metric depth estimates. Let us briefly analyze what happens in the image domain, where the projection of the profile cut $\mathcal{C}$ is supposed to go through corresponding pixels in the two views. Fig. 9 shows the zoom of a region in the stereo pair of example (a) in Fig. 8. The DP estimation leads to the best matching results, proving that SymStereo can achieve accuracies of 1-2 pixels for an image resolution of 1280×960 whenever the surface is textured.

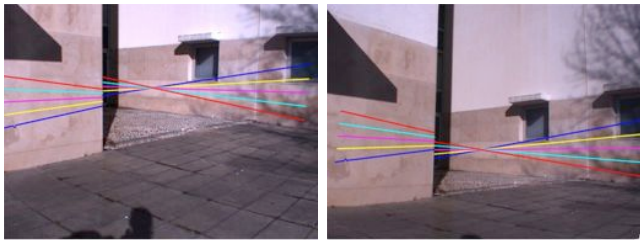
[2] Note that all parameters of the algorithm are kept constant for all the experiments.



(a) Top Image ($\mathtt{I}_1$)

(b) Bottom Image ($\mathtt{I}_2$)

Fig. 9. Image matches obtained in case (a) of Fig.8 for the region outlined in yellow. As in the previous examples, (green) corresponds to the LRF measurements, (red) is $\mathcal{C}_{DP}$, and (blue and cyan) is $\mathcal{C}_{MRF}$, where in the case of (cyan) the MRF decided for the DP label. In the case the virtual scan plane intersects a textured region or near strong edges, the matching obtained from dynamic programming over E is very accurate.



(a) Top Image ($\mathtt{I}_1$)

(b) Bottom Image ($\mathtt{I}_2$)

Fig. 10. Multi-cut example. The correspondence between contours in the top and bottom images can be idenfied by the color coding.

It is also interesting to observe that the projection of the range data obtained with LRF is slightly off in terms of stereo correspondence. This is explained by small errors in the extrinsic calibration between the cameras and the LRF that can hardly be avoided.

Finally, Fig. 10 shows that our algorithm enables independent depth estimate along multiple virtual scan planes, with the only constraint being that the scan planes must intersect the stereo baseline. The fact that SRF can mimic multiple 2D LRFs is an important benefit of the described framework that can be decisive for many robotic projects.

VI. Discussion and Conclusion

We presented a new stereo pipeline specially tailored for estimating depth along a virtual scan plane. The approach, named Stereo Rangefinding (SRF), was experimentally compared against LRF in several examples representative of different possible operation conditions. The results were encouraging in terms of showing that passive stereo can be leveraged to meet the robustness and depth accuracy of laser range data. SRF proved to be as accurate as LRF in most of the tests, but important issues remain for the case of the profile cut lying in slanted surfaces with very low texture. Nevertheless, it is reasonable to claim that the current version of the algorithm can be an alternative to LRF for operating at short-medium ranges in man-made environments.

An important aspect that has not be discussed is the computational complexity of the algorithms and the possibility of achieving real-time performance. Since the work is still in an exploratory stage, the experimental results were obtained using straightforward implementations in Matlab, with runtimes close to 1 minute per stereo pair. However, we are convinced that computational complexity will not be a major issue, and that it will be possible to achieve rates of several frames per second. The underlying reason is the fact that, with the exception of the MRF labeling, all the steps in the pipeline have deterministic runtimes that can be easily accelerated using GPGPU techniques. Moreover, the MRF formulation is very simple and the convergence time is usually small.

VII. Acknowledgments

Michel Antunes acknowledges the Portuguese Science Foundation (FCT) that generously funded his work through the grant SFRH/BD/47488/2008. This work was also supported by the Portuguese Science Foundation (FCT) under the project grants PTDC/SEN-TRA/099413/2008 and PTDC/EEA-AUT/113818/2009.

References

[1] Yoshiro Negishi, Jun Miura, and Yoshiaki Shirai. Mobile robot navigation in unknown environments using omnidirectional stereo and laser rangefinder. In *Int. Conf. on Intelligent Robots and Systems*, 2004.

[2] Cristiano Premebida, Oswaldo Ludwig, and Urbano Nunes. Lidar and vision-based pedestrian detection system. *J. Field Robotics*, 2009.

[3] Fabio Ramos Bertrand Douillard, Dieter Fox. Laser and vision based outdoor object mapping. In *Robotics: Science and Systems*, 2008.

[4] Fabio T. Ramos, Juan I. Nieto, and Hugh F. Durrant-Whyte. Recognising and modelling landmarks to close loops in outdoor slam. In *ICRA*, 2007.

[5] Srikanth Saripalli, James F. Montgomery, and Gaurav S. Sukhatme. Visually-guided landing of an unmanned aerial vehicle. *Trans. on Robotics and Automation*, 2003.

[6] Scharstein, D., Szeliski, R., and R. Zabih. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. In *International Journal of Computer Vision*, 2002.

[7] Michel Antunes, Joao P. Barreto, and Xenophon Zabulis. Plane surface detection and reconstruction using induced stereo symmetry. In *British Machine Vision Conference*, 2011.

[8] Michel Antunes and Joao P. Barreto. Stereo estimation of depth along virtual cut planes. In *Int. Conf. on Computer Vision Workshop (CVVT)*, 2011.

[9] Morales, Sandino, Klette, and Reinhard. Ground truth evaluation of stereo algorithms for real world applications. In *ACCV 2010 Workshops*, 2011.

[10] Hrabar Stefan. An evaluation of stereo and laser-based range sensing for rotorcraft unmanned aerial vehicle obstacle avoidance. *Journal of Field Robotics*, 2012.

[11] Jean Yves Bouguet. Camera calibration toolbox for matlab. `http://www.vision.caltech.edu/bouguetj/calib_doc/index.html`.

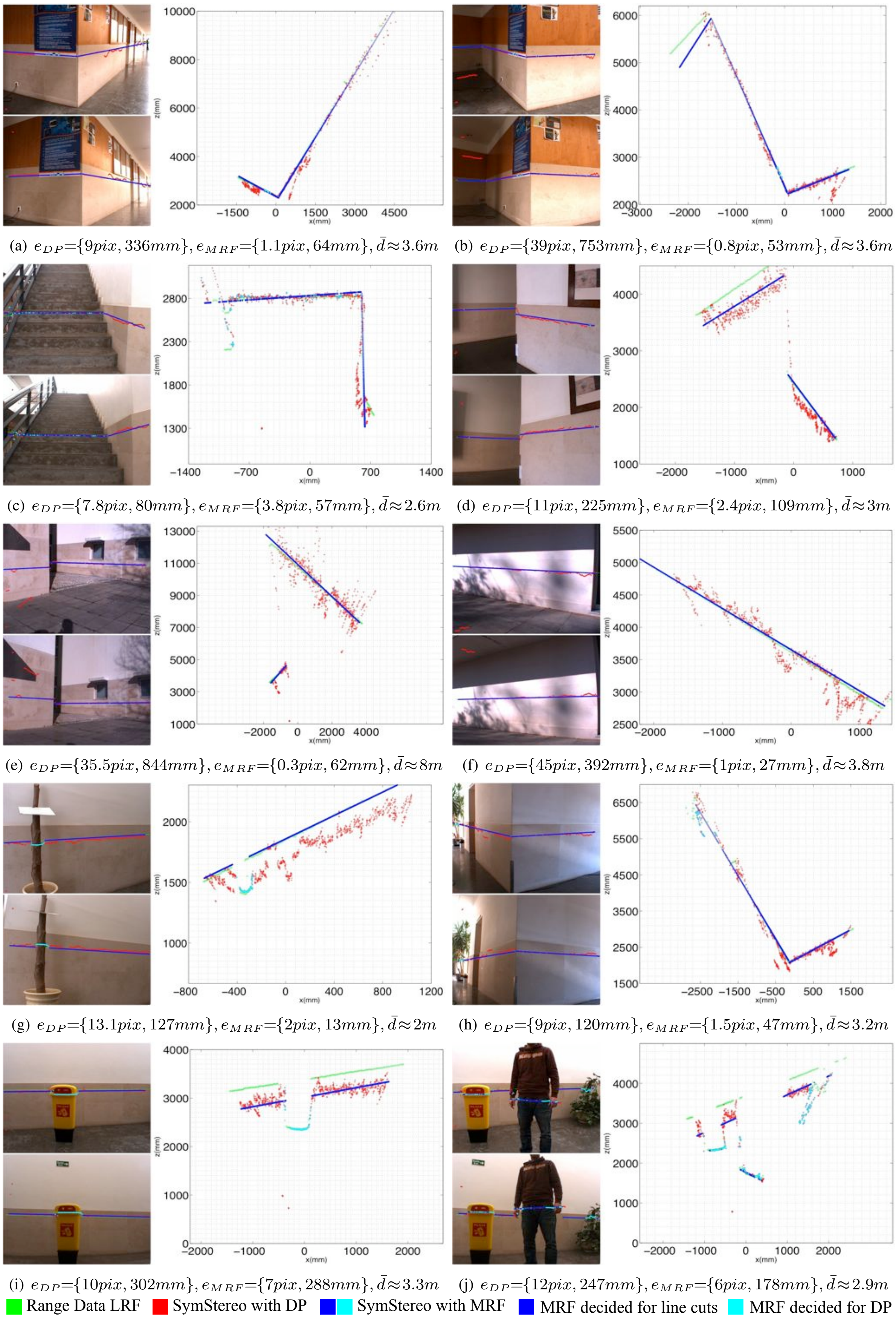
[12] Francisco Vasconcelos, Joao P. Barreto, and Urbano Nunes. A minimal solution for the extrinsic calibration of a camera and a laser rangefinder. *Trans. on Pattern Analysis and Machine Intelligence*, 2011.

[13] Robert T. Collins. A space-sweep approach to true multi-image matching. In *Computer Vision and Pattern Recognition*, 1996.

[14] Peter Kovesi. Symmetry and asymmetry from local phase. In *Tenth Australian Joint Converence on Artificial Intelligence*, 1997.

[15] Yuri Boykov and Vladimir Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *Trans. on Pattern Analysis and Machine Intelligence*, 2001.

[16] Michel Antunes and Joao P. Barreto. Plane surface detection and reconstruction using induced stereo symmetry. *Technical Report*, 2011.



(a) e_{DP}=$\{9pix, 336mm\}$, e_{MRF}=$\{1.1pix, 64mm\}$, $\bar{d}\approx 3.6m$

(b) e_{DP}=$\{39pix, 753mm\}$, e_{MRF}=$\{0.8pix, 53mm\}$, $\bar{d}\approx 3.6m$

(c) e_{DP}=$\{7.8pix, 80mm\}$, e_{MRF}=$\{3.8pix, 57mm\}$, $\bar{d}\approx 2.6m$

(d) e_{DP}=$\{11pix, 225mm\}$, e_{MRF}=$\{2.4pix, 109mm\}$, $\bar{d}\approx 3m$

(e) e_{DP}=$\{35.5pix, 844mm\}$, e_{MRF}=$\{0.3pix, 62mm\}$, $\bar{d}\approx 8m$

(f) e_{DP}=$\{45pix, 392mm\}$, e_{MRF}=$\{1pix, 27mm\}$, $\bar{d}\approx 3.8m$

(g) e_{DP}=$\{13.1pix, 127mm\}$, e_{MRF}=$\{2pix, 13mm\}$, $\bar{d}\approx 2m$

(h) e_{DP}=$\{9pix, 120mm\}$, e_{MRF}=$\{1.5pix, 47mm\}$, $\bar{d}\approx 3.2m$

(i) e_{DP}=$\{10pix, 302mm\}$, e_{MRF}=$\{7pix, 288mm\}$, $\bar{d}\approx 3.3m$

(j) e_{DP}=$\{12pix, 247mm\}$, e_{MRF}=$\{6pix, 178mm\}$, $\bar{d}\approx 2.9m$

🟩 Range Data LRF 🟥 SymStereo with DP 🟦🟦 SymStereo with MRF 🟦 MRF decided for line cuts 🟦 MRF decided for DP

Fig. 8. Qualitative comparison between the profile cuts $\mathcal{C}$ estimated using SRF and the measurements provided by a LRF. (Green) - Measurements of the LRF, (red) $\mathcal{C}_{DP}$ estimated using Dynamic Programming, (blue and cyan) labeling obtained from the MRF, where (blue) are points to which a line segment was assigned, while for the (cyan) points the MRF decided for the non-straight ($\mathcal{C}_{DP}$). We also show the average distances e_{DP} and e_{MRF} (final estimation) between the range data provided by the LRF and the points in $\mathcal{C}_{DP}$ and C_{MRF}, respectively (the first value is measured in the reference image, while the second value concerns depth measurements). $\bar{d}$ represents the average distance of the LRF readings from the origin.