An Integrated Framework for Feature Extraction, Object Recognition and Stereo Vision with GPU support

Alexander Woodward, Patrice Delmas*
Dept. of General Systems Studies, University of Tokyo, 3-8-1 Komaba, Tokyo, 153-8902, Japan
Email: alex@sacral.c.u-tokyo.ac.jp
*Dept. of Computer Science, The University of Auckland, Auckland 1142, New Zealand

***Abstract*—This paper investigates the integration of feature extraction, object recognition and 3D reconstruction by stereo vision into a unified framework. In doing so, stereo vision can be made more robust by applying feature extraction results to the stereo matching process, and object recognition can be extended through the integration of depth information as another feature of the scene. In this work a hierarchical feature extraction algorithm using Gabor filters is combined with a multi-path dynamic programming stereo algorithm. Subsequently, from this combination a new matching cost measured is proposed. The design is suitable for implementation on the graphics card (GPU), making it a target for real-time computer vision. This paper focuses on the framework's application to stereo reconstruction and experiments show its ability to robustly match regions and preserve object depth boundaries through the use of feature analysis.**

I. Introduction

The goal of this work is to investigate the integration of computer vision processes for 3D reconstruction by stereo vision, feature extraction, object recognition into a unified framework. In doing so, stereo vision can be made more robust by applying feature extraction results to the stereo matching process, and conversly, object recognition can be extended through the integration of depth information as another feature of the scene.

Design decisions were driven by the desire to have a parallel-ready solution that would be suitable for implementation on the graphics card (GPU) - this will be completed and described in future work[1].

Our experiments have shown that stereo matching can be performed on 640×480 pixel images in excess of ≈ 85 fps on current hardware[2]. This means that computational resources can be assigned to additional analysis of images while maintaining real-time rates. Both the chosen hierarchical feature extraction model and stereo matching algorithm are both easily parallelisable and it is hoped that a perceptual framework for real-time computer vision can be formalised from this work.

This paper outlines the components of the framework, with first analysis focusing on its stereo reconstruction properties. The hierarchical feature extraction model is introduced in Sec. III. Section IV describes the stereo matching algorithm and Sec. V describes how the two models can be combined to define a new and robust cost function for stereo matching. A set of results and analysis are given in Sec. VI.

II. Related Work

Feature extraction is a first step for computer vision algorithms requiring higher order information about a scene for tasks such as object recognition. Common low-level features are pixel intensities, colour information, edges, corners, lines, circles etc., features such as SIFT and SURF are also popular [1].

This work investigates a hierarchical feature extraction model based upon the work of Mutch et. al [2] which follows a developmental line of successful, biologically inspired object recognition models, collectively referred to as the HMAX model. Its basic principal is the repeated application of Gabor filters at various orientations on an image at different scales. At higher levels of the model the strongest responses are pooled together in order to gain a level of scale invariance. This model was chosen due to its suitability for implementation on the graphics card (GPU) as it involves many independent, low-level computations across the image.

Stereo correspondence algorithms have been investigated in works such as Woodward et al. [3] and the well known Middlebury stereo website [4]. Many commercial stereo vision systems share a common trait in only using very simple stereo correspondence algorithms running in hardware: the Focus Robotics' PCI nDepth Vision System [5] and the Point Grey Research's Bumblebee Stereo Vision Camera System [6] both use the sum of absolute differences (SAD) approaches. These examples reflect the computational burden imposed by dense stereo correspondence algorithms - therefore this work focuses on a model that can leverage the parallelism of the GPU to provide real-time stereo vision and feature extraction. For GPU stereo, [7], [8] are good examples – other hardware platforms such as stereo vision on FPGAs has been investigated in [9]. However, the benefits of GPUs lie in their cheaper cost, ease

[1] Currently only the stereo vision component of the proposed framework has been implemented on the GPU.

[2] Tested on an Intel Core i7-2600 Quad Core 3.4 Ghz, 16 Gb RAM and Nvidia GeForce GTX580 graphics card

of programming and compilation allowing for code flexibility, and the rate at which newer and faster cards appear on the market; a new generation of card appears roughly every year. As mentioned in the introduction, real-time rates are easily achieved and we are now in a position where additional image analysis can be conducted on the GPU alongside a stereo vision algorithm while maintaining real-time operation.

A number of works have investigated combining feature extraction with stereo vision. As examples, Lysak et. al [10] investigated using Gabor filters to determine strong seed points to initialise stereo matching - in our work Gabor filters are instead used for the cost function and a more complete feature extraction model that can be used for object recognition is also looked at. Trapp et. al [11] proposed using Gabor filters as a cost measure in a similar manner, but within a paradigmatically different stereo algorithm not targeted for GPU implementation.

III. Hierarchical Feature Extraction Model - HMAX

The chosen model involves the convolution of Gabor filters at different orientations with an input image at various scales, along with local maxima pooling of strong filter responses to acquire a level of scale invariance. Given a training image set, these features can then be sampled and used to construct a dictionary for object recognition purposes - running an unseen image through the model and comparative measurements can be taken between the input and the dictionary of features by using a classification algorithm (e.g. support vector machine (SVM)). This model has been successfully used for object recognition in a number of works, exemplified in Mutch et. al [2].

The model has a feedforward design and each progressive filtration level is as follows (and as depicted in Fig. 1):

1) $S1$: Gabor filter responses at different orientations and image scales.
2) $C1$: a local maxima pooling operation to the Gabor filter responses at different scales. This step finds the strongest scale-invariant response at an image point to a certain filter orientation.
3) $S2$: a metric comparison between the $C1$ level and a set of dictionary features.
4) $C2$: a local maxima pooling operation over scales of responses to dictionary features.

Each filtration level consists of a number of image scalings, described as a level's layers. Between all levels a common real valued *retinal* coordinate system is defined so that the output of different filter responses can be easily related between layers. This common retinal coordinate system is important for being able to give input values from a lower level filter to a higher level filter, or to determine an image's entire set of feature vectors defined at the base (finest) image resolution.

Figure 1 gives a more detailed description of the HMAX model:

1) W, H are the base image's dimensions, S is the number of scale sizes (number of responses at a particular filtration level), RI is the raw input image, SI are scalings of the raw image,
2) The $S1$ level is the result of applying F_1 numbers of Gabor filter orientations at S scales.
3) The $C1$ level is the local maxima pooling operation.
4) The $S2$ level is the result of a comparison between the $C1$ layer and a feature dictionary set of size F_2.
5) Finally the $C2$ level is a local maxima pooling of the $S2$ level which can be used with a standard classification algorithm.

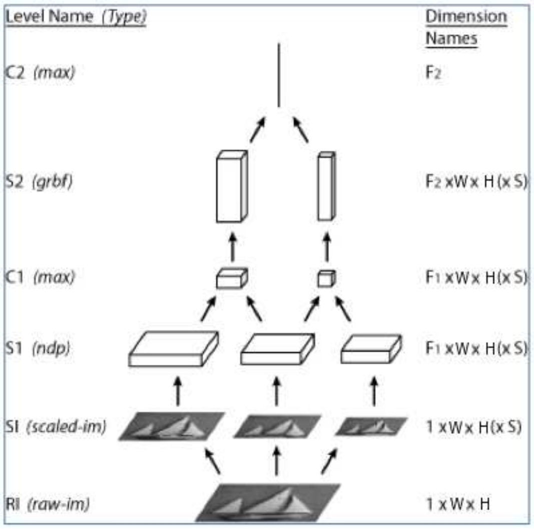


Fig. 1. The full feature extraction model depicting the levels needed for an object recognition task [2]. For stereo matching we are only interested up to the $C1$ level. Each level has a number of layers for each scaling. Please refer to the main body (Sec. III) for the full description of this diagram.

For stereo vision we want to construct a feature space, $F(x, y, k)$, at the raw input image scale. The determination of k, the number of features for a particular pixel position in the base image, is the sum of the number of Gabor filter responses at different scales from the $S1$ layer, in addition to the number of local maxima responses from the $C1$ layer - for stereo vision we are only interested in using the information contained up to the $C1$ layer, the filtration levels $S2$, $C2$ are only used for object recognition tasks. Without the local maxima operation, the feature vector at $F(x, y, *)$ is equivalent to a *Gabor Jet*, which describes the local spectral content about a pixel position.

A. Gabor Filter Definition

A Gabor filter can be considered as the composition of a plane wave multiplied by a 2D Gaussian function. It is mathematically defined in the complex plane but for this work only the real component is used:

$$G(\mathbf{p}, \lambda, \theta, \sigma, \gamma) = \exp(-\frac{\acute{x}^2 + \gamma^2 \acute{y}^2}{2\sigma^2}) \cos(2\pi \frac{\acute{x}}{\lambda}) \qquad (1)$$

with $\acute{x} = x \cos\theta + y \sin\theta$ and $\acute{y} = -x \sin\theta + y \cos\theta$.

Gabor filters are truncated to 11×11 pixels in size[3], therefore x and y vary between -5 and 5 (local image regions defined by windows of size $2n+1 \times 2n+1$ where $n=5$). Here, θ varies between 0 and π, $\gamma = 0.3$ defines the aspect ratio, $\sigma = 4.5$ defines the effective width of the filter, and $\lambda = 5.6$ defines the wavelength [2]. It should be noted that scaling the input images and running filters of a fixed size is equivalent to keeping the image size constant and applying filters of different scales.

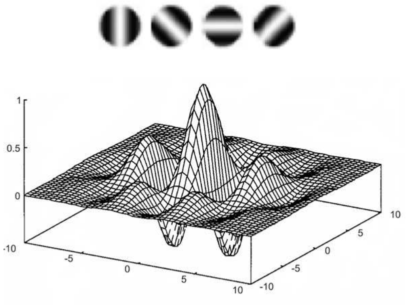


Fig. 2. A Gabor filter visualisation and four orientations used in this work.

For a local image region X and Gabor filter G, consisting of pixels $i \in \{0, ..., 11 \times 11\}$, the normalised filter response R is given by:

$$R(X,G) = \left| \frac{\sum X_i G_i}{\sqrt{\sum X_i^2}} \right| \tag{2}$$

IV. Stereo Vision using the Semi-Global Matching (SGM) Algorithm

Binocular stereo involves the recovery of depth information from a pair of cameras viewing the same scene. Consequently, a stereo algorithm involves the identification of conjugate points in the stereo image pair. Through only the parallax of conjugate points one can computationally evaluate the depth over a scene.

The most suitable approach for real-time stereo is an algorithm that can be easily scaled up in quality when faster hardware becomes available. Therefore, we chose to implement the Semi-global matching (SGM) algorithm, first proposed by Hirschmüller [12]. This algorithm is based around multiple 1D dynamic programming optimisations in different scans through the disparity cost volume. Dynamic programming without back-tracing has a very small memory footprint and only requires the previous disparity column to be stored in local memory as the algorithm progresses.

For computational efficiency, stereo correspondence assumes the standard stereo geometry (SSG) setup with coplanar image planes, this is achieved by performing a stereo image rectification on calibrated cameras, as described in [13].

A. Pixelwise Cost Calculation

A disimmilarity measure (cost) $C(x,y,d)$ is taken between each pixel grey-value at $\mathbf{p} = (x,y)$ in the base image, $I_b(\mathbf{p})$, and at $\mathbf{q} = (x-d, y), d \in [d_{min}, d_{max}]$ in the match image, $I_m(\mathbf{q})$. This measure is taken as the sum of dissimilarities within local matching windows (of size $M \times N$) around $\mathbf{p}$ and $\mathbf{q}$, appropriate sizes were empirically found to be in the range $M, N \in [1, 15]$. Tested measures included the Birchfield and Tomasi sampling insensitive cost measure C_{BT} , and the sum of absolute pixelwise differences (SAD), $C_{SAD}(x,y,d)$. The computationally light SAD measure was chosen for this work.

B. SGM Optimisation Step

For a particular scan direction $\mathbf{v}$, the optimised cost $L_{\mathbf{v}}(\mathbf{p}, d)$ for a pixel position $\mathbf{p}$ and disparity d is recursively given as:

$$\begin{aligned} L_{\mathbf{v}}(\mathbf{p}, d) = C(\mathbf{p}, d) + \min(&L_{\mathbf{v}}(\mathbf{p}-\mathbf{v}, d), \\ &L_{\mathbf{v}}(\mathbf{p}-\mathbf{v}, d-1) + P_1, \\ &L_{\mathbf{v}}(\mathbf{p}-\mathbf{v}, d+1) + P_1, M_{\mathbf{p},\mathbf{v}} + P_2) - M_{\mathbf{p},\mathbf{v}} \end{aligned} \tag{3}$$

where $M_{\mathbf{p},\mathbf{v}} = \min_i L_{\mathbf{v}}(\mathbf{p}-\mathbf{v}, i)$ is the minimum matching cost for the previous pixel position, $\mathbf{p}-\mathbf{v}$. The regularisation parameters, P_1 and P_2 ($P_1 \leq P_2$), are set with respect to local matching window size since pixel-wise costs are summed. Costs $L_{\mathbf{v}}$ are summed over directional scans through the cost volume:

$$S(\mathbf{p}, d) = \sum_{i=1}^{n} L_{\mathbf{v_i}}(\mathbf{p}, d) \tag{4}$$

where n is the number of scan directions and the upper limit for S is $S \leq n(C_{max} + P_2)$, here C_{max} can be set to an arbitrary 'large' value, dependent on an implementation's primitive data type. Finally, the disparity for pixel $\mathbf{p}$ can be chosen by taking the minimal aggregated cost of the column $S(\mathbf{p}, *)$ - doing this for all pixels generates the scalar disparity map $D(x,y)$, $d \in [d_{min}, d_{max}]$.

Occlusions can be found by comparing disparity maps generated using the costs from matching the base image to the match image, D_b, and costs from match to base, D_m. The final disparity map, D, can be marked with invalid disparities, $d_{invalid}$, if the two conjugate disparity values from both maps differ by a threshold ϕ:

$$D(\mathbf{p}) = \begin{cases} D_b(\mathbf{p}) & \text{if } |D_b(\mathbf{p}) - D_m(\mathbf{q})| < \phi \\ d_{invalid} & \text{otherwise} \end{cases} \tag{5}$$

The computational complexity of the algorithm is $O(WHd_{range})$ [12], where W, H are the dimensions of the input images and $d_{range} = d_{max} - d_{min}$ is the disparity range. Here, the number of optimisation passes and local matching window size are the parameters that most influence computation time. Regularisation parameters P_1 and P_2 control how smooth the disparity volume should be and act to remove noise. When $P_1 = P_2 = 0$ the algorithm functions as a simple

[3] It has been noted that there is no gain in classification performance for larger arrays of the same filter [2].

winner takes all (WTA) approach. With a single optimisation pass along scanlines, SGM performs as a traditional dynamic programming stereo algorithm. This scalability allows a wide generation of GPUs to be supported.

Our implementation of SGM supports up to 8 passes and on an Intel Core i7-2600 Quad Core 3.4 Ghz, 16 Gb RAM and Nvidia GeForce GTX580 graphics card, the algorithm was capable of $\approx$ 25 fps with 8 passes and $\approx$ 85 fps with a single optimisation pass for test images of 640×480 pixels.

V. New Cost Measure for Stereo Matching using HMAX Feature Vectors

First let $C_{HMAX}(x,y,d)$ define the pairwise absolute difference between feature vectors, F_l and F_r (as described in Sec. III) summed for a local image window:

$$C_{HMAX}(x,y,d) = \sum |F(x,y,k) - F(x-d,y,k)| \quad (6)$$

The complete cost measured is formed by adding the $C_{SAD}(x,y,d)$ cost measure as follows:

$$C_F(x,y,d) = \alpha C_{HMAX}(x,y,d) + (1-\alpha)\cdot s\cdot C_{SAD}(x,y,d) \quad (7)$$

This final measure has an adjustable weighting coefficient, $\alpha \in [0,1]$, for a tradeoff between the original matching cost and the additional information provided by the hierarchical feature model, here s is an empirically chosen normalisation factor.

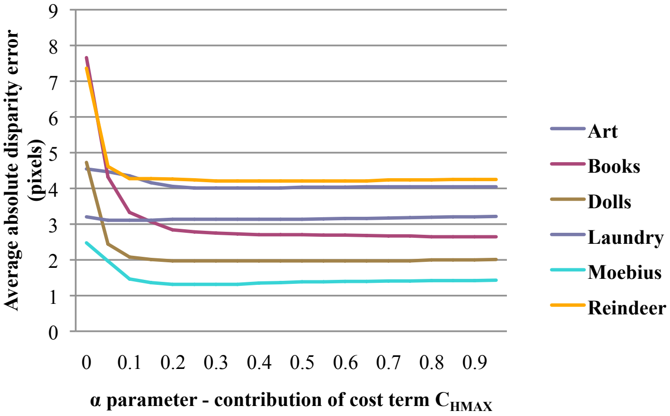


Fig. 3. Analysis of improvement made by the new cost measure C_F for a set of stereo images with ground truth from the Middlebury 2005 stereo dataset [4]. The plot shows the improvement (decrease in absolute disparity errors, measured in pixels, between generated disparity and ground truth) made by gradually increasing the contribution of the HMAX feature response cost, (C_{HMAX}, as α is increased from 0 to 1 (refer to Equ. 7).

VI. Results

Figure 4 presents results of the proposed framework when applied to the stereo correspondence problem. A stereo pair consisting of strong occluded regions can be difficult to reconstruct, Fig. 4a shows an example of this situation where a stereo image of a test subject has been taken (256×256 pixel images). Stereo matching was performed on this pair using the same settings, $P_1 = 1, P_2 = 10$ and the number of features $k = 13$, while only altering the cost function being used. When the original cost function, C_{SAD}, is used the algorithm handles textureless regions well but at the same time regularises over occluding boundaries - Fig. 4e. The new cost $C_F(x,y,d)$, given in Equ. 7 with $a = 0.4$, manages to preserve boundaries while retaining good regularisation over homogeneous image regions - Fig. 4f. Fig.4d shows improved reconstruction for the test subject's face region using the new cost measure - a zoom in of this region in Fig. 4e and Fig. 4f shows this more clearly.

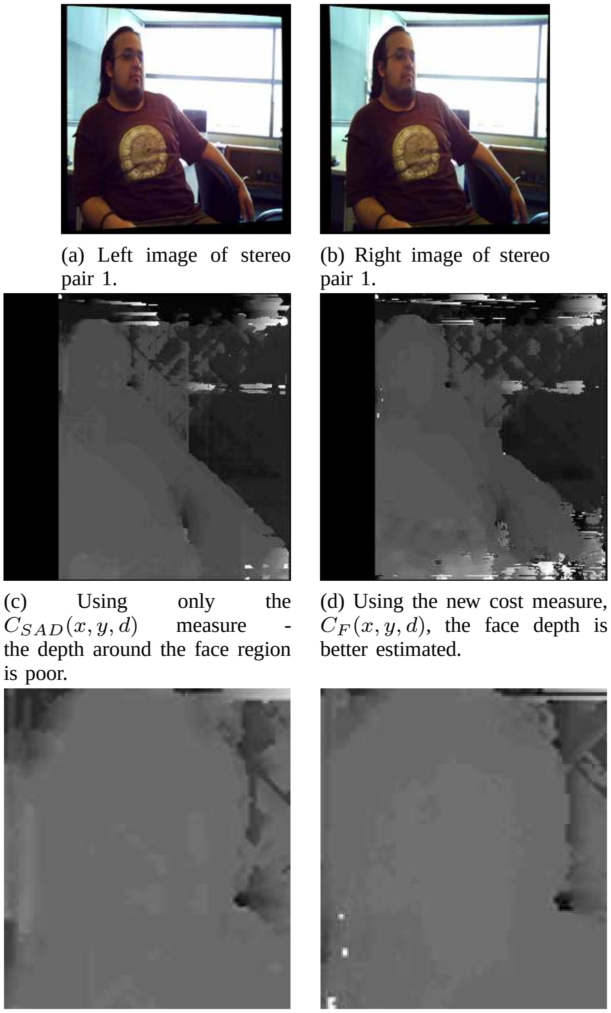
Figure 3 shows the quantitative improvement made by the new cost measure C_F (Equ. 7) applied to a stereo dataset obtained from the Middlebury website (the 2005 Stereo datasets with ground truth) [4]. Here the contribution of the HMAX features (C_{HMAX}) is gradually increased by adjusting the contribution parameter α in Equ. 7. The average absolute disparity difference, measured in pixels, between the stereo reconstruction and the provided ground truth were measured for different values of α. The most noticeable improvement is between not using the HMAX features ($\alpha = 0$) to ≈ 0.25 when the plots for all image test sets show a consistent improvement. For all test sets, when $\alpha > 0.25$ the accuracy remains stable and does not change, this points to the HMAX features containing the same intrinsic information as what the MSE measure provides, making the MSE component unnecessary after a point.

VII. Conclusion and Future Work

This paper has proposed an integrated framework for feature extraction and stereo vision. In doing so, a new stereo matching cost measure has been created which, when used with the SGM stereo algorithm, is capable of improving the robustness of depth reconstruction of a scene. Design choices were made with a view for parallelisation and future work looks at a complete GPU implementation. The SGM algorithm has been implemented on the GPU and as an example of reduction in computation time, a greater than 90 fold speed increase over the CPU implementation on an Intel Core i7 960 @ 3.2 GHz was observed using the CUDA API and a NVIDIA GTX 470 graphics card.

Now that a framework has been proposed, further analysis into the best choice of number of Gabor and local maxima poolings should be investigated. An analysis on their individual weightings within the feature vector would determine what tradeoffs in computation time and matching accuracy can be made. Intuitively, a feature analysis should cover a range of scales and orientations to obtain enough information to uniquely identify local pixel regions.

As a next step, an investigation into how depth information can be used for feature matching within this framework will be pursued. This requires calibration of the cameras to obtain metric information instead of relative disparity values. A step to remove positional invariance in the depth volume would



(a) Left image of stereo pair 1.

(b) Right image of stereo pair 1.

(c) Using only the $C_{SAD}(x, y, d)$ measure - the depth around the face region is poor.

(d) Using the new cost measure, $C_F(x, y, d)$, the face depth is better estimated.

(e) A zoom in of the face region. Using only the $C_{SAD}(x, y, d)$ measure.

(f) A zoom in of the face region, using the new cost measure, $C_F(x, y, d)$. Despite being low resolution, there is a better delineation of the face region using the new measure.

Fig. 4. Example visualisation. The test stereo image has a low disparity range but using the new cost measure C_F helps give more robust depth estimates. Comparing Fig. 4e and Fig. 4f one can see a larger region of greater disparity instead of only the head shape being delineated.

also need to be investigated, akin to the local maxima pooling operation over scales in the HMAX model.

References

[1] Open source BSD license, "OpenCV computer vision library," Open source BSD license., 2011. [Online]. Available: http://sourceforge.net/projects/opencvlibrary/. [Accessed: Sep. 14, 2011].

[2] J. Mutch and D. G. Lowe, "Object class recognition and localization using sparse features with limited receptive fields," *International Journal of Computer Vision (IJCV)*, vol. 80, no. 1, pp. 45–57, October 2008.

[3] A. Woodward, D. An, Y. Lin, P. Delmas, G. Gimel'farb, and J. Morris, "An Evaluation of Three Popular Computer Vision Approaches for 3-D Face Synthesis," in *Proceedings of Structural, Syntactic, and Statistical Pattern Recognition, (SSPR)*, China, 2006, pp. 270–278.

[4] D. Scharstein, "Middlebury Stereo Vision Page," 2007. [Online]. Available: http://vision.middlebury.edu/stereo/. [Accessed: Sep. 15, 2011].

[5] Focus Robotics, "PCI nDepth Vision System," 2008. [Online]. Available: http://www.focusrobotics.com/products/systems.html. [Accessed: Sep. 14, 2010].

[6] Point Grey Research, Inc., "Bumblebee Stereo Vision Camera System Datasheet," 2011. [Online]. Available: http://www.ptgrey.com/products/bumblebee2/bumblebee2_xb3_datasheet.pdf. [Accessed: Sep. 14, 2011].

[7] A. Brunton, C. Shu, and G. Roth, "Belief propagation on the gpu for stereo vision," in *Proceedings of the 3rd Canadian Conference on Computer and Robot Vision (CRV06)*, 2006, pp. 76–76.

[8] I. Ernst and H. Hirschmuller, "Mutual information based semi-global stereo matching on the gpu," in *Proceedings of the International Symposium on Visual Computing (ISVC)*, 2008, pp. I: 228–239.

[9] K. J. J. Morris and G. L. Gimel'farb, "Intelligent vision: A first step - real time stereovision," in *Proceedings of the Conference on Advanced Concepts for Intelligent Vision Systems (ACIVS)*, 2009, pp. 355–366.

[10] Y. Lysak and O. Kapshiy, "A dense stereo matching by iterated dynamic programming with using gabor filters," in *Modern Problems of Radio Engineering, Telecommunications and Computer Science, 2008 Proceedings of International Conference on*, feb. 2008, pp. 348 –349.

[11] R. Trapp, S. Dre, and G. Hartmann, "Stereo matching with implicit detection of occlusions," in *Computer Vision ECCV98*, 1998, pp. 17–.

[12] H. Hirschmller, "Accurate and efficient stereo processing by semi-global matching and mutual information," in *Proceedings of the Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE Computer Society, 2005, pp. 807–814.

[13] Y. Lin, A. Woodward, D. An, J. Morris, P. Delmas, G. Gimel'farb, and J. Morris, "Rectifying Images for Stereo Vision," in *Proceedings of Image and Vision Computing New Zealand Conference (IVCNZ)*, New Zealand, 2006, pp. 13–17.