

FAST MODEL BASED STEREO MATCHING USING SOAR

Yusuf Ozturk** and Arvind Sridharan*
San Diego State University
Electrical and Computer Engineering
yozturk@mail.sdsu.edu**, sridhara@rohan.sdsu.edu*

ABSTRACT

Stereo correspondence is maybe the most prominent step in extraction of three dimensional structure of a scene from two or more images taken from distinct viewpoints. The correspondence problem consists of determining the locations in each image that are projections of the same physical point in space. This paper introduces a novel model based stereo matching algorithm using System of Associative Relations (SOAR) computational model. SOAR makes use of pair-wise pixel interactions to determine the underlying structure of associations within the token. The proposed stereo correspondence algorithm utilizes feature vectors (tokens) formed by direction of derivatives which constitute SOAR feature vectors. The algorithm proposed here is simple and easily realizable in hardware. Proposed algorithm is observed to yield equivalent matching performance at lower computation cost.

1. INTRODUCTION

Computational stereo deals with extraction of three dimensional structure from two or more images of the same scene. The research in the computational stereo goes as early as 1970s with several milestone developments to date. A through review of most recent advances in stereo correspondence methods and real time implementations is provided in [10] by Brown et al. The stereo matching or correspondence problem concerns with matching points or other primitives in two or more images such that the matched points are projections of the same point in the scene. The fundamental basis for stereo is the fact that a single three dimensional physical location projects to a unique pair of image locations in two observing cameras [10]. Given two images collected from two cameras, the goal is to find, for every pixel or primitive in left image, the corresponding point or primitive in the right image.

In Figure 1, the spatial distance between the projection of point P onto left and right images is named as the disparity of that point after being converted to metric units. The difference values between the pixel positions matched to each other are stored in an image forms the disparity map.

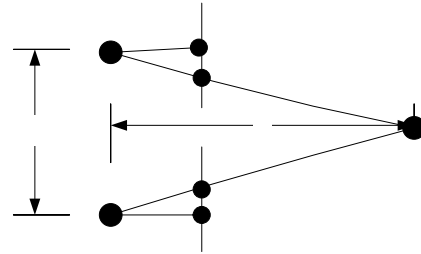


Figure 1: Geometry of stereo matching

Given the focal length f , of the camera and the distance T between the cameras, the distance Z of the point P from the optical center of the two cameras can be computed as given in eq. (1).

$$Z = f \frac{T}{d} \quad \text{where } d = x_l - x_r \quad [1]$$

In the literature both local and global matching constraints for mapping the two projections of a three dimensional physical point has been addressed. Area based stereo matching algorithms compare intensity similarity of pixels within a window centered at pixel (x, y) in left image with a window centered around the pixel at location $(x + d, y)$ in the right image[1,6,5,7]. The value of d for which the two windows are most similar is the disparity of the point (x, y) and thus the object containing point (x, y) . This approach makes a reasonable assumption that a pixel is surrounded by a window of pixels, which have approximately same disparity. This area based method, which assumes that there is a large enough intensity variation within a window also requires that intensity variation is small enough not to cross disparity discontinuities, is very popular due to its ease of implementation. Most area based methods also employ some or all of the following constraints:

- *Epipolar constraint* requires that all matching points lie on the corresponding epipolar lines of the two images.
- *Uniqueness constraint* enforces the match to be unique between the two images.
- *Smoothness constraint* enforces the disparity map to be locally smooth.

- *Ordering constraint* enforces points from the one image to occur in the same order on the epipolar line in the other image.

Area based stereo mapping algorithms suffer from the dependency on the intensity variation and usually result in poor results compared to feature based stereo matching algorithms. Several stereo matching algorithms using wavelets [2], phase [3] and filter based [4, 8] feature sets have been reported in the literature. The area based stereo matching algorithms suffers from gray level dependency where feature based algorithms suffers from computational complexity. Here we propose a stereo matching algorithm that does not depend solely on gray level averages and yet it is simple and easily realizable in hardware. The algorithm we propose here is based on the recently developed System of Associative Relations (SOAR) [9] architecture and utilizes feature vectors (Tokens) formed by direction of derivatives. SOAR makes use of pair-wise pixel interactions to determine the underlying structure of associations.

2. SYSTEM OF ASSOCIATIVE RELATIONS

The System of Associative Relations (SOAR) makes use of the pair-wise pixel associations. The pixels defined by a token over an image are used to determine associations among pixels. This analysis token is basically an irregular mask that will determine the underlying structure of associations. The size and shape of the token will depend on the target application. The token can simply be a 3x3 or a 5x5 window centered at a pixel whose associations are explored. It can be of any shape such as a T or a circular window. Two possible token patterns are given in Figure 2.

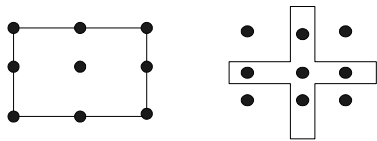


Figure 2: Possible token shapes

It is worth to mention that SOAR does not assume a particular token shape and size; it can be readily used for other token shapes. To explore the associations among pixel elements defined by the token SOAR maps each pixel to a processing node and sets the internal status of this processing node to be equal to the intensity of that particular pixel.

Suppose pixel intensities in a block (token) are : $\{ V_{ij}; i,j=1,2,\dots,q \}$. SOAR encodes the larger/smaller

information called the “interpixel connection strength” between pixels in this block as the signum of differences as

$$T(i,j;k,l) = \text{sgn}\{V_{ij}, V_{kl}\} \quad \text{where } 1 \leq i, j, k, l \leq q \quad (2)$$

except: $(i,j) \neq (k,l)$

The signum function here is the ternary sign function of discrete mathematics.

In addition to formulating a prescription to encode inter-pixel relationships in a block, an ensemble inter-pixel association over P number of blocks is also formulated in [9] as given in eq.(3).

$$T_p(i,j;k,l) = \sum_{p=1}^P \text{sgn}(V_{ij}^p - V_{kl}^p) \quad (3)$$

In eq.(2) the connection strength $T(i,j;k,l)$ between the pixel at location (i,j) and the one at (k,l) increases if $V_{ij} > V_{kl}$; but it decreases if $V_{ij} < V_{kl}$, and remains unchanged if $V_{ij} = V_{kl}$. Furthermore, we would like to point out that the connection strengths are anti-symmetric, i.e., $T(k,l;i,j) = -T(i,j;k,l)$.

Suppose a token has been placed at any position in a given image and the pixel associations between pixels have been determined using (2). In this case, we can define the similarity measure among the initial token which has been stored in connections $T(i,j;k,l)$ and a new token at another location by:

$$E = \sum_i \sum_j \sum_k \sum_l T(i,j;k,l) * \text{sgn}(V_{ij} - V_{kl}) \quad (4)$$

If we use (2) for the term $T(i,j;k,l)$ in (3) we can write:

$$E = \sum_i \sum_j \sum_k \sum_l \text{sgn}(V_{ij}^1 - V_{kl}^1) * \text{sgn}(V_{ij} - V_{kl}) \quad (5)$$

It is not difficult to deduce from eq. (4) that E will attain its maximum value if the two patterns are correlated with a correlation coefficient of “1.” But it will reach its minimum value if the two patterns are correlated with a correlation coefficient of “-1.” Finally, E will have a value near zero if the two patterns are not correlated. That is, there is no resemblance between ordering of the pixels within the tokens compared. It can also be seen that the similarity measure given in eq. (5) is a *correlation indicator function* between two specific patterns.

This similarity measure is analogous to correlation in the case of single pattern comparisons. As stated above the SOAR similarity measure attains its maximum value when the two compared patterns have a similar ordering of the pixels. In addition, we would like to point out that the SOAR similarity measure is independent of the mean gray

level of the tokens. That is, it does not matter how much bias exists in the data, the ordering between pixels for similar textures will be the same.

3. STEREO MATCHING USING SOAR

A solution to the correspondence problem in stereo vision can be implemented using SOAR. The degree of association of pixels within any arbitrary token of the image can be used as the parameter to obtain a match between the left and right images.

In this paper, we assume that the matching points lie on corresponding epipolar lines (scanline) of the stereo pair images. We also assume that the images are grayscale. A suitable transformation might be necessary if the images do not meet these assumptions. The left image is assumed to be the reference image and the images are scanned from left to right and top to bottom. Furthermore, the token for obtaining the associations is assumed to be a 3x3 square though other shapes may also be used as mentioned earlier.

The algorithm starts from a point on the left image, $L(x, y)$, and searches for the best match in the right image, $R(x + d, y)$, along the same scanline within a search depth as shown in Figure 3. This match is obtained by finding the most similar tokens of the left and right images using eq. (5). This process is applied to successive points along the scanline and all scanlines in the image and moving the window centered on the point in question.

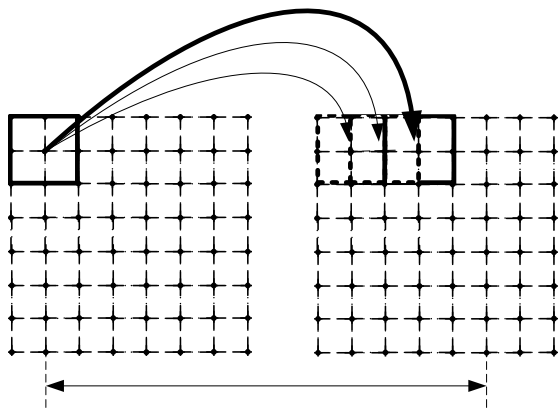


Figure 3: Left and Right images with $R(x + d, y)$ being most similar to $L(x, y)$

As in most area based algorithms, the photometric properties of the image are used for the evaluating the similarity function given by eq. (4). False matches arising from noise, occlusion and distortion can be minimized by

applying the uniqueness constraint by recovering from previous erroneous matches as long as there are better matches available within the search depth s as the algorithm proceeds.

3.1 SOAR Stereo Correspondence Algorithm

Inputs to the algorithm:

- * Left and right grayscale images
- * Search depth s
- * Size and shape of the window

For each horizontal scanline in the left image:

- * Obtain the pixel association (given by eq. 2) within the window centered on the pixel in sequence.
- * Starting at the corresponding pixel in the right image obtain pixel association centered on that pixel and find the best match (given by eq. 5) within the search depth s .
- * Store disparity for pixel.
- * Plot disparity map.

4. SIMULATION RESULTS

We have implemented the SOAR stereo matching algorithm with several token sizes and shapes. The disparity values in each case is mapped to gray levels for visualization.

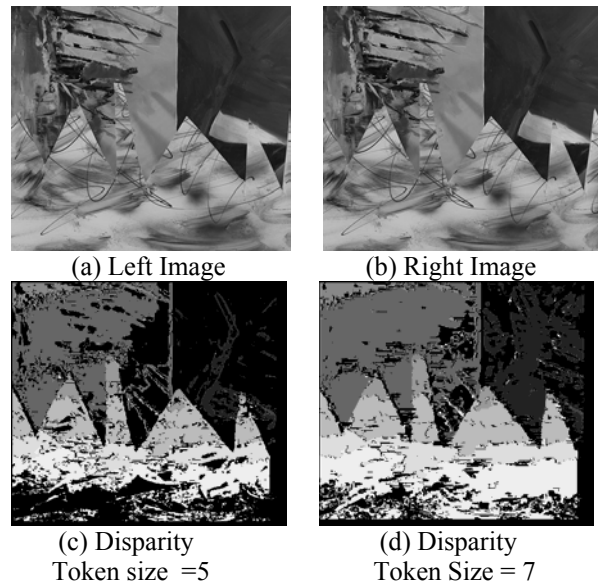
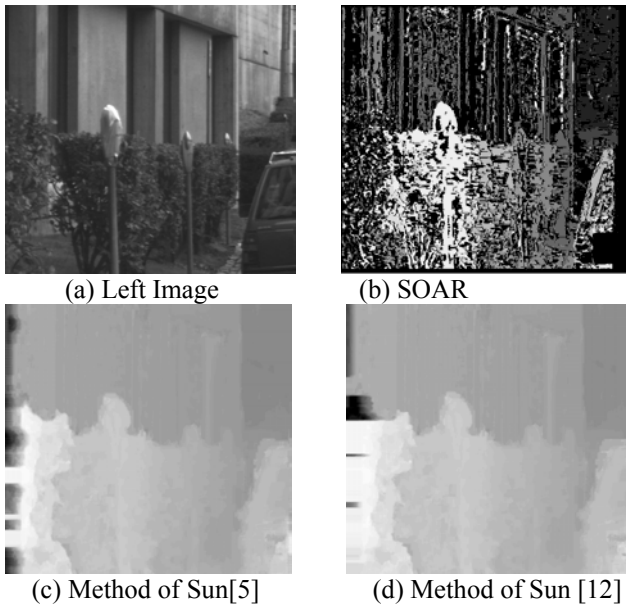
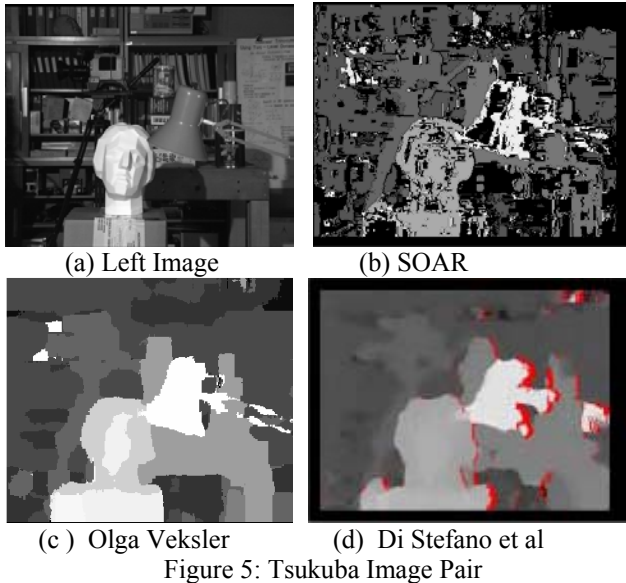


Figure 4: Sawtooth image pair

In Figure 4, we present the sawtooth stereo image pair together with the disparity images for token sizes of 5 and seven respectively. As it is expected the larger token size

results in better estimation of the Disparity Map due to the increased information content in a larger token. Both token size 5 and token size 7 are implemented to satisfy uniqueness and ordering constraints.

In Figure 5, we studied the Tsukuba image pair and provided the outcome of proposed algorithm along with the results reported by Velsker[6] and Di Stefano[11]. Results clearly indicates that SOAR stereo matching algorithm yields comparable results to those of Velster and Di Stefano at reduced complexity.



In Figure 6, we have studied the park image and compared our results with those reported by Sun in [5] and [12]. The proposed algorithm again yielded similar performance to

that of Sun. Please note that we did not employ smoothing to reflect the true performance of the SOAR stereo matching algorithm.

5. REFERENCES

1. Birchfield, S. and Tomasi, C, "Depth discontinuities by pixel by pixel stereo", *International Journal of Computer Vision*, 35(3):269-293.
2. Kim, Y-S., Lee, J.J., and Ha, Y. H., "Stereo matching algorithms based on Modified Wavelet decomposition process", *Pattern recognition*, 30(6):929-952.
3. Porr, B., Cozzi, S., and Worgotter, F. "How to hear visual disparities: real time stereoscopic spatial depth analysis using temporal resonance", *Biological cybernetics*, 78(5):329-336.
4. Jones, D.G. and Malik, J., "Computational framework for determining stereo correspondence from a set of linear spatial filters", *Image and Vision Computing*, 10(10):699-708.
5. Changming Sun, "Fast stereo matching using rectangular subregions and 3D maximum surface techniques", *International Journal of Computer Vision*, Vol.47, No.1/2/3, 99-117, May 2002.
6. Olga Veksler, "Stereo correspondence with compact windows via minimum ratio cycle", *IEEE transactions on patterns analysis and machine intelligence*, Vol. 24, No. 12, December 2002.
7. C. Lawrence Zitnik, and Tekeo Kanade, "A cooperative algorithm for stereo matchin and occlusion detection", *IEEE patterns analysis and machine intelligence*, Vol. 22, no: 7, July 2000.
8. Candocia F., Adjouadi M., "A similarity measure for stereo feature matching", *IEEE transactions on image processing*, Vol. 6, No. 10, October 1997.
9. Y. Ozturk, H. Abut, "SOAR: System of Associative Relations", *Signal Processing: Image Communications* Vol. 17, 2002, pp. 261-276.
10. M.Z. Brown, D. Burschka, G.D. Hager, "Advances in Computational Stereo", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 25, No. 8, August 2003.
11. L. Di Stefano, M Marchionni, S Mattoccia, G Neri, "A Fast Area-Based Stereo Matching algorithm", 2002.
12. Changming Sun, "A fast stereo matching method", *Digital Image Computing: Techniques and Applications*, pages 95-100, Massey University, New Zealand.