

Multi-level Dynamic Programming for Axial Motion Stereo Line Matching

Raymond K K Yip
Department of Electronic Engineering
City University of Hong Kong
83, Tat Chee Avenue, Kowloon, Hong Kong.
email: eeryyip@cityu.edu.hk

ABSTRACT

In this paper, a multi-level dynamic programming approach is used to solve the line segment based correspondence problem in axial motion stereo. In this method, a *Local Similarity Measure* is calculated for each line segment pair between the *Front* and *Back* images. In level 1, the *matching probability* between line segments is represented by their *Local Similarity Measure (LSM)*. Line segment pair that have a *matching probability* larger than a threshold T_1 is selected as potential matching pair. T_1 is set to a relative high value so that the probability of correct match of level 1 is very high. Dynamic programming is then used to search for their best match. Based on the geometric properties between the matched and the unmatched line segments, a *Global Similarity Measure (GSM)* is calculated for each unmatched line segment pair. An overall *Similarity Measure (matching probability)* is then obtained by the *LSM* and the *GSM*. Then, the algorithm begin the second match but with a slightly lower threshold T_2 . The new matched results are then used to modify the *GSM* and the overall *Similarity Measure*. These processes are repeated until a predefined level n_{stop} (or a predefined condition) is reached. By using the *GSM* and multi-level searching technique, the proposed technique increases the matching accuracy and reduce the number of unmatched line segment due to misordering when dynamic programming is used for axial motion stereo matching.

1. INTRODUCTION

One of the major area in computer vision is the recovery of 3D information (depth map) using stereo vision analysis. In stereopsis, one of the most important and difficult stage is stereo matching. It is the process of identifying the corresponding 2D features between images that are belongs to the same physical point in the 3D scene. This problem is also known as the *correspondence problem*.

The most popular cameras arrangement for multiple images is the lateral stereo model [2, 4-6]. Recently, researchers have been moved towards studying images of a moving camera system. Axial motion stereo [1, 3] is one of the technique in which the camera system is moved along its optical axis.

Many lateral stereo matching techniques have been proposed in the past decades [2, 4-6]. A review [2] is provided by U. R. Dhond and J. K. Aggarwal. One of the important approach is the use of dynamic programming [2, 4, 5]. It has been proved to be one of the simple and efficiency technique for stereo matching. In this paper, a line segment based multi-level dynamic programming method is proposed to solve the corresponding problem of axial motion stereo images from a single camera. The proposed technique increases the matching accuracy and reduce the number of unmatched line segment when dynamic programming is used for axial motion stereo matching.

2. MULTI-LEVEL LINE MATCHING PROCESS

In this paper, the corresponding problem of axial motion stereo is formulate as:

- 1) Assign unique labels to each line segment of the front F (f_i) and back B (b_j) images.
- 2) Calculate a *Local Similarity Measure*, $LS(f_i, b_j)$, for each line segment pair (f_i, b_j).
- 3) Use $LS(f_i, b_j)$ as the overall *Similarity Measure*, $SM(f_i, b_j)$, of (f_i, b_j) and select a threshold T .
- 4) Select those line segment pairs whose $SM(f_i, b_j)$ is greater than T .
- 5) Use dynamic programming to find the best match for those selected from step 3.
- 6) Use the matched results to obtain a *Global (Structure) Similarity Measure*, $GS(f_i, b_j)$, for each unmatched line segment pair (f_i, b_j).
- 7) Calculate the $SM(f_i, b_j)$ by $LS(f_i, b_j)$ and $GS(f_i, b_j)$ for each unmatched line pair.
- 8) Lower the threshold T and repeat step 4 to 8 for all unmatched line segments until a predefined condition is meet.

The proposed method is a multi-level line matching process with the first level matching those line segments that have a very high similarity value ($SM(f_i, b_j) > T$). Thus, the probability of correct match in this level is very high which provides a strong basis (accurate references) to assist the matching process of the next level for weaker matches. The matching criterion is gradually decreases (by lowering T) as the level increases so stronger matches are matched first then followed by weaker matches. Furthermore, matched line segments in all previous levels are used to assist the matching process of the next level so as to improve the chances of correct match of weaker matches.

2.1 Ordering, Constraints and Similarity Measure

In each level (level n), unmatched line segments in the front image F and back image B are sorted in ascending order w.r.t. x -coordinates of their mid-points. For those points that have the same x -coordinates, their y -coordinates are used to sorted the order between them.

If any line segment b_j in B satisfied all the following constraints, b_j is a potential matching line of f_i and a similarity measure will be calculated to represent the similarity between these two lines.

1. Focus of Expansion (FOE) Constraint and FOE Similarity Measure

As the camera only moves along the optical axis. The displacement vector between correspondence features will intersect at a point known as the focus of expansion (FOE) [1, 3], see figure 1.

In this paper, f_i and b_j are considered as satisfying the FOE constraint

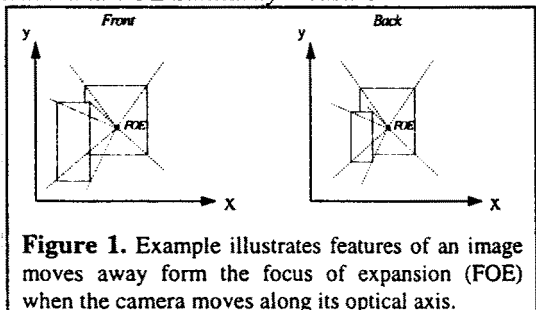


Figure 1. Example illustrates features of an image moves away from the focus of expansion (FOE) when the camera moves along its optical axis.

$$\text{if } S_{FOE}(f_i, b_j) > T_n \quad (1)$$

$$\text{with } S_{FOE}(f_i, b_j) = e^{\frac{-|n_dist| \cdot S_j}{\text{Theoretical_len}(f_i, b_j)}} \quad (2)$$

and $S_j = 4$, n_dist is the normal distance between the mid-point of b_j to the line formed by the mid-point of f_i and FOE (Figure 2), $\text{Theoretical_len}(f_i, b_j)$ is the length of the equation formed by b_j overlapped with the triangle formed by the end points of f_i and FOE (figure 3), and T_n is the threshold of level n .

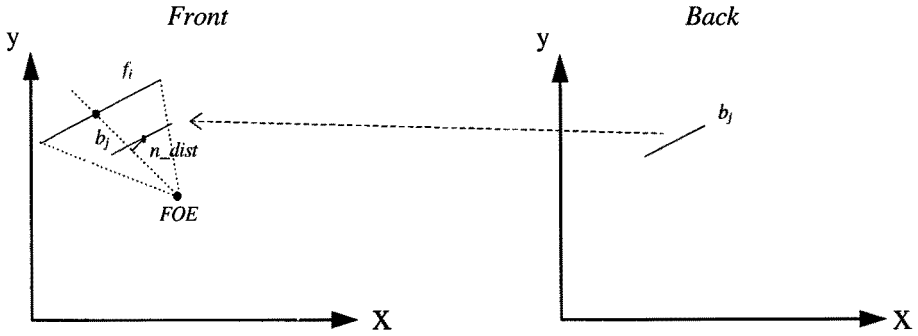


Figure 2. n_dist is the normal distance between the mid-point of b_j to the line formed by the mid-point of f_i and FOE.

2. Overlap Similarity Constraint

f_i and b_j are considered as overlapping

$$\text{if } S_{OLP}(f_i, b_j) > T_n \quad (3)$$

$$\text{with } S_{OLP}(f_i, b_j) = \frac{\min[\text{Real_len}(f_i, b_j), \text{Theoretical_len}(f_i, b_j)]}{\max[\text{Real_len}(f_i, b_j), \text{Theoretical_len}(f_i, b_j)]} \quad (4)$$

and $\text{Real_len}(f_i, b_j)$ is the length of b_j overlapped with the triangle formed by f_i and FOE, $\text{Theoretical_len}(f_i, b_j)$ is the length of the equation formed by b_j overlapped with the triangle formed by the end points of f_i and FOE (figure 3), and T_n is the threshold of level n .

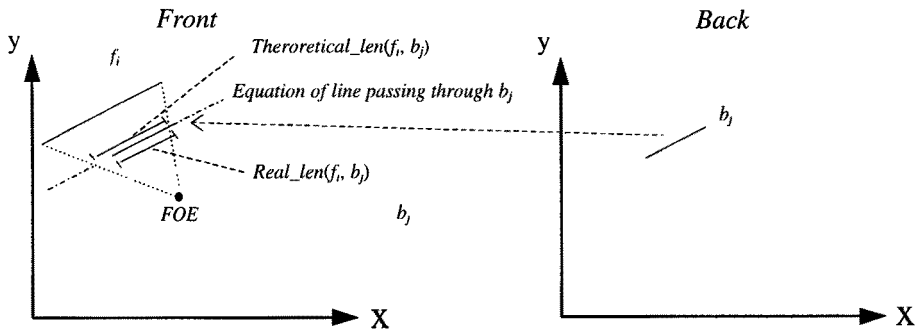


Figure 3. Example showing the $\text{Real_len}(f_i, b_j)$ and $\text{Theoretical_len}(f_i, b_j)$ of the overlap similarity constraint.

3. Orientation Constraint and Orientation Similarity Measure

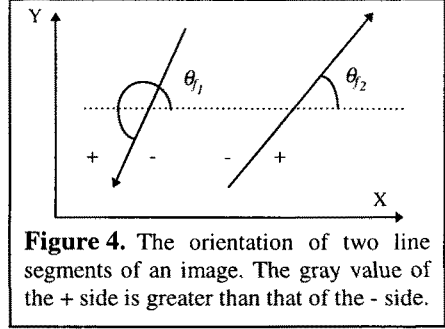
The orientations of f_i and b_j are considered as similar

$$\text{if } OS(f_i, b_j) > C_{OS} \quad (5)$$

$$\text{with } OS(f_i, b_j) = \left[\cos(\theta_{f_i} - \theta_{b_j}) \right]^{S_2} \quad (6)$$

and $C_{OS} = 0.9$, $S_2 = 4$.

The definition of orientation of a line segment is shown in Figure 4.



4. Global (Structure) Similarity Constraint and Measure

Once when matching pair(s) is/are obtained, a factor $GS(f_i, b_j)$ is added in the next level to measure the *Global (Structure) Similarity* between f_i and b_j . In addition, a factor $SE(f_i, b_j)$ is used to calculate the *Global (Structure) error* between f_i and b_j . If $SE(f_i, b_j)$ is larger than a threshold, then $GS(f_i, b_j)$ is set to 0. The structure of f_i and b_j are considered as similar

$$\text{if } GS(f_i, b_j) > T_n \quad (7)$$

$$\text{and } SE(f_i, b_j) < 0.2k \quad (8)$$

$$\text{with } GS(f_i, b_j) = \frac{1}{2k} \sum_{p=1}^k (e^{-|\Delta \cos m\theta_p| S_3} + e^{-|\Delta \cos m\phi_p| S_3}) \quad (9)$$

$$SE(f_i, b_j) = \sum_{p=1}^k \delta_p \quad (10)$$

$$\delta_p = \begin{cases} 1 & \text{if } \Delta \cos m\theta_p > 0.2 \text{ or } \Delta \cos m\phi_p > 0.2 \\ 0 & \text{otherwise} \end{cases} \quad (11)$$

$$\Delta \cos m\theta_p = \cos m\theta_{f(p, i)} - \cos m\theta_{b(p, i)} \quad (12a)$$

$$\Delta \cos m\phi_p = \cos m\phi_{f(p, i)} - \cos m\phi_{b(p, i)} \quad (12b)$$

and $S_3 = 10$; T_n is the threshold of level n ; k is the number of all matched pairs; $\cos m\theta_{f(p, i)}$ is defined as the horizontal angle between the mid-points of f_p and f_i ; $\cos m\theta_{b(p, j)}$ is defined as the horizontal angle between the mid-points of b_p and b_j ; $\cos m\phi_{f(p, i)}$ is defined as the vertical angle between the mid-points of f_p and f_i ; $\cos m\phi_{b(p, j)}$ is defined as the vertical angle between the mid-points of b_p and b_j ; $\Delta \cos m\theta_p$ is defined as the horizontal angles difference between (f_p, f_i) and (b_p, b_j) ; $\Delta \cos m\phi_p$ is defined as the vertical angles difference between (f_p, f_i) and (b_p, b_j) ; f_p and b_p are matched pair obtained from all previous level(s).

Figure 5 illustrates this idea.

From equation (9) and figure 5, it can be seen that the $GS(f_i, b_j)$ measures the angles difference between the potential matching pair (f_i, b_j) with all matched (f_p, b_p) line segments. Equation (11) defines the maximum allowed angle deviation (0.2) for each

(f_p, f_i) and (b_p, b_i) . If either the horizontal or vertical angle difference between (f_p, f_i) and (b_p, b_i) are larger than the maximum allowed angle deviation, (f_p, f_i) and (b_p, b_i) is considered as a mismatched case (i.e. $\delta_p = 1$). If the total number of mismatched cases of (f_i, b_j) (equation (10)) is larger than 20% of the number of matched line segments (equation (8)), (f_i, b_j) is considered as mismatched and $GS(f_i, b_j)$ will be set to 0.

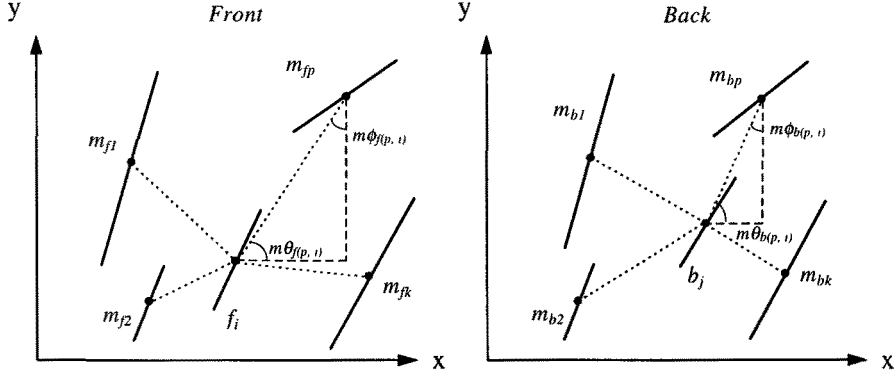


Figure 5. The relationship of line segments f_i and b_j with all previous matched results $m_{f1}, m_{f2}, \dots, m_{fp}, \dots, m_{fk}$ and $m_{b1}, m_{b2}, \dots, m_{bp}, \dots, m_{bk}$ respectively.

2.2 Matching with Dynamic Programming

A multi-stage weighted search graph [3][5] G_n (level n) is then constructed after the above process. A primitive G_1 (practically, the graph is more complicate) is shown in Figure 5. A null vertex $v_{i,0}$ is added to each stage V_i to represent the condition of no matching occurs between the front image and the back image. All possible path must start from stage s and terminate in t . And each vertex $v_{i,j}$ of the weighted graph G_n has an associated weight $SM(f_i, b_j)$ to indicate the *Similarity Measure* between f_i and b_j ,

$$SM(f_i, b_j) = \begin{cases} \frac{w_1 S_{FOE}(f_i, b_j) + w_2 S_{OLP}(f_i, b_j) + w_3 OS(f_i, b_j) + w_4 GS(f_i, b_j)}{\sum_{a=1}^4 w_a} & \text{if } (f_i, b_j) \text{ satisfies all constraints} \\ 0 & \text{otherwise} \end{cases} \quad (13)$$

with $w_1 = 1, w_2 = 1, w_3 = 1$ and $w_4 = 6$.

An accumulated weight $ASM(v_{i,j})$ is used in dynamic programming to search the maximal weighted path of G_n . It is defined as

$$ASM(v_{i,j}) = \max_{\forall v_{i-1,p} \in V_{i-1}} [ASM(v_{i-1,p}) + SM(l_i, r_j)] \quad (14)$$

and the maximal path is back traced from stage V_t to V_s . The matched result is stored in M which contains the index pairs $[(m_{l1}, m_{r1}), (m_{l2}, m_{r2}), \dots, (m_{lk}, m_{rk})]$ of line segments.

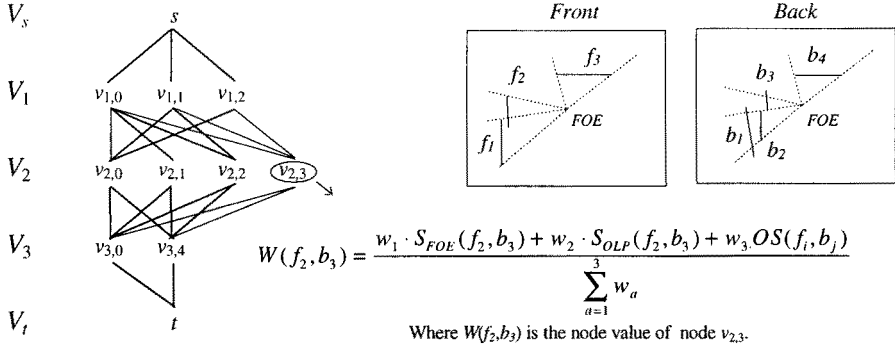


Figure 5. A primitive weighted graph G_I is shown where all possible paths must start from s and terminate at t .

2.3 Update the Similarity Measure and Threshold Decreasing

The matched results of each level is used to update the $GS(f_i, b_j)$ of all the unmatched potential matching pairs. A lower threshold T_n is then used for the next level. The process of matching, updating and lowering threshold is repeated until a predefined level n_{stop} (or a predefined condition) is reached.

By using this multi-level (repeating) searching technique, errors due to broken lines, partially occlusion and misordering are greatly reduced. Figure 6 shows the problem of misordering between the front and back images. Figure 6 indicates the first two line segments of the front and back images are labelled in reverse order. A one pass dynamic programming method will result in either (f_1, b_2) , (f_3, b_3) , (f_4, b_4) or (f_2, b_1) , (f_3, b_3) , (f_4, b_4) . But for the proposed multi-level searching technique, the remaining pair will be matched in the next level or some level afterwards.

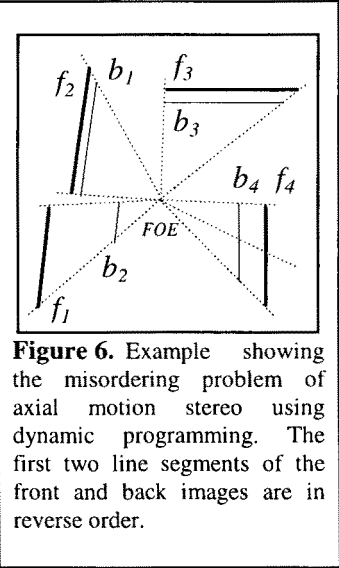


Figure 6. Example showing the misordering problem of axial motion stereo using dynamic programming. The first two line segments of the front and back images are in reverse order.

3. EXPERIMENTAL RESULTS AND DISCUSSION

In this paper, an image pairs are used to demonstrate the capacity of the proposed method. An imperfect line segment detector is used to extractor line segments form the images, line segments shorter than 4 pixels are eliminated from the results. The value of T_n in each level and experimental results are summarized in Table 1. For comparison, results of applying a one pass dynamic programming method are shown in Table 2. Experiments are done with the threshold T being set to 0.5 and 0.0.

Figure 7 shows the results of matching a block world. Figure 7 (a) and (b) show the origin front and back images. Figure 7 (c) and (d) show the final results with the total number of matched pairs = 113 and 101 are correctly matched; successful rate = 89.38%.

Results of the experiment show that the proposed multi-level dynamic programming method achieves a good successful rate under complex environments. Compare its results with the one pass method, it can be seen that the proposed method improved both the number of correctly matched line segments and the successful rate of the one pass dynamic programming.

From Table 2, it can be observed that although lowering the threshold value T (from 0.50 to 0.00) improves the number of correct matches of the one pass approach, it also increases the number of mismatched and lower the accuracy (successful rate) of the overall result. On the other hand, the multi-level method increases the number of correctly matched line segments but with a fairly constant successful rate. Furthermore, as the threshold of the one pass method is already set to 0 and the multi-level method can obtained a higher number of correctly matched lines, it can be concluded that some of the unmatched line segments of the one pass method are due to misordering; and the multi-level approach does reduce the number of unmatched line segments due to misordering when dynamic programming is used.

In this paper, we presented a multi-level dynamic programming method for solving the correspondence problem of axial motion stereo. Experimental results show the feasibility of the proposed method in handling these situations. By using the *Global Similarity Measure* and multi-level searching technique, the proposed technique increases the matching accuracy and reduce the number of unmatched line segment due to misordering when dynamic programming is used for axial motion stereo matching.

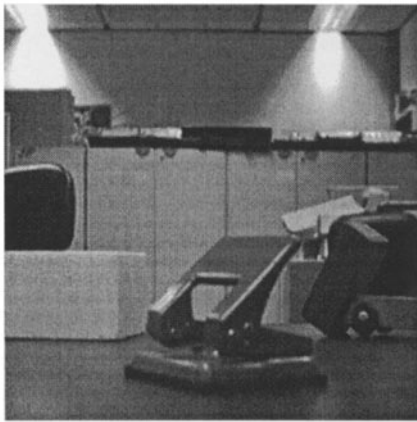
Table 1. Experimental results, Figure 7, Blocks - $F = 175$ lines and $B = 192$ lines.									
		Matched lines							
		In each level			Accumulated				
n	T_n	Correct	Wrong	Total	Correct	Wrong	Total	Failure rate	Successful rate
1	0.90	4	0	4	4	0	4	0.00%	100.0%
2	0.80	16	2	18	20	2	22	9.09%	90.91%
3	0.70	25	3	28	45	5	50	10.00%	90.00%
4	0.60	15	1	16	60	6	66	9.09%	90.91%
5	0.50	15	1	16	75	7	82	8.54%	91.46%
6	0.40	11	1	12	86	8	94	8.51%	91.49%
7	0.30	6	1	7	92	9	101	8.91%	91.09%
8	0.20	4	3	7	96	12	108	11.11%	88.89%
9	0.10	2	0	2	98	12	110	10.91%	89.09%
10	0.00	3	0	3	101	12	113	10.62%	89.38%

Table 2. Experimental results of the one pass dynamic programming approach.						
		Matched lines				
	T	Correct	Wrong	Total	Failure rate	Successful rate
Blocks	0.50	57	7	62	11.29%	88.71%
	0.00	91	13	104	12.50%	87.50%

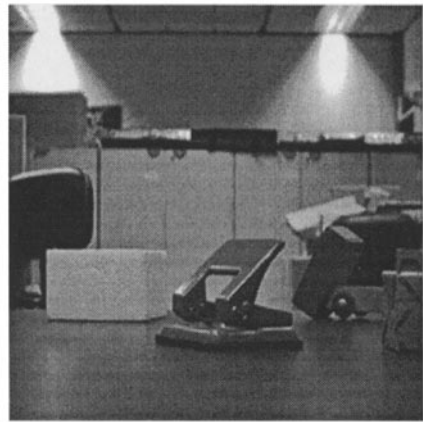
REFERENCES

- [1] Nicolas Alvertos, Dragana Brzakovic and Rafael C. Gonzalez, "Camera Geometries for Image Matching in 3-D Machine Vision", *IEEE Trans. on PAMI*, Vol. 11, No. 9, pp. 897-915, Sept. 1989.

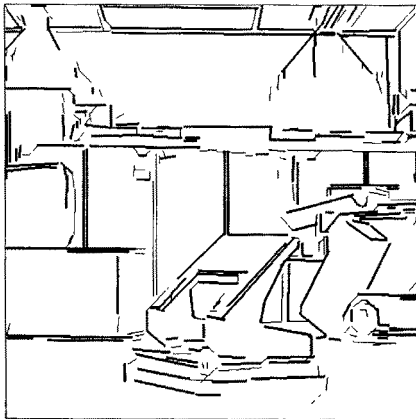
- [2] U.R. Dhond and J.K. Aggarwal, "Structure from stereo - A Review", *IEEE Trans. Syst. Man Cybern.*, Vol. 19, No., 6, pp. 1489-1510, Nov./Dec. 1989.
- [3] X.Y. Jiang and H. Bunke, "Line Segment Based Axial Motion Stereo", *Pattern Recognition*, Vol. 28, no. 4, pp. 553-562, 1995.
- [4] S. H. Lee and J. J. Leou, "A dynamic programming approach to line segment matching in stereo vision", *Pattern Recognition*, vol. 27, no. 8, pp. 961-986, 1994.
- [5] Z. N. Li, "Stereo correspondence based on line matching in Hough space using dynamic programming", *IEEE Trans. Syst. Man Cybern.*, vol. 24, no. 1, pp. 144-152, 1994.
- [6] R.K.K. Yip and W.P. Ho, "Multi-level Based Stereo Line Matching with Structure Information Using Dynamic Programming", *Proceeding of the 1996 International Conference on Image Processing*, Lausanne, Switzerland, September 16-19, 1996.



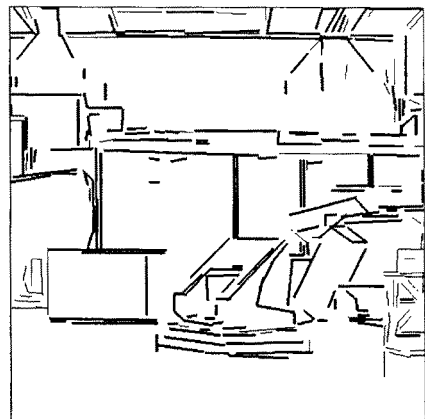
(a) Front image of a block world.



(b) Back image of a block world.



(c) Final matched results.



(d) Final matched results.

Figure 7. Matched results of a block world. Total number of match = 113, correct match = 101, successful rate = 89.38%.