

Enhanced Image Feature Extraction for Object Tracking*

Dimitrios Charalampidis

Vesselin P. Jilkov

Department of Electrical Engineering
University of New Orleans, New Orleans, LA 70148, USA
{dcharala, vjilkov}@uno.edu, (504) 280-{7415, 3950 (fax)}

Abstract— *A feature-based technique is presented for 3D motion and structure estimation of rigid objects from a sequence of images. Considered is a feature-based rigid object representation which is more complete as compared to previous techniques. In this approach the features provide information regarding the location and type of significant object components and connections. A robust enhanced Hough transform approach is proposed for line and junction detection. Based on a perspective projection imaging model the 2D descriptive and relational features are used for estimating the 3D motion of a rigid body. The proposed technique is evaluated for various objects and image sequence scenarios.*

Keywords: Tracking, 3D motion estimation, feature extraction

1 Introduction

This paper is concerned with estimating the *3D motion* and *structure* of a rigid object from a long sequence of 2D monocular images. In contrast to the customary point-target tracking, where the motion of some target “center” is of interest¹, this problem involves estimating the rotational motion of the object body as well as some of its structure parameters². Within the standard state-space estimation framework, two main approaches have been applied to this problem — feature-based [1], [2] and optical flow-based [3]. The optical-flow based approach relies on measuring (or computing) the velocity of the points in the image plane. In this work we pursue the feature-based approach. The target is considered as a set of features which satisfy the rigid body assumption — all feature points do not change their relative

positions. The 2D features are obtained from raw 2D images by a signal processing algorithm referred to as *feature extractor*. Feature correspondences (*feature matching*) within multiple-frames need to be established in order to estimate the 3D object motion, based on assumed 3D motion and 2D feature “measurement” models. The latter part of the problem—estimating the object state (motion and structure parameters) based on feature-point measurements — has been well developed in both batch (parameter estimation) [2], [4], [5] and recursive (Kalman or nonlinear) filtering [1], [6], [5] [7], [8], [9] frameworks. All of above cited works, however, assume perfect feature extraction and matching and deal with the estimation issues only. In reality the feature extraction and matching are crucial for the performance of an overall estimation algorithm. While abundant publications on feature extraction (e.g., [10], [11], [12]) and feature matching [13] exist in the image processing literature it is rarely the case they are considered in an integrated dynamic feature detection-tracking framework. Apparently, a great potential for enhancement of the feature extraction exists, as compared to a “static” (single frame) feature extraction, if the valuable information from the state estimator (predictor) is used. The same argument is valid for the feature matching, as well. The ultimate aim of our work is development of an overall 3D object tracker based on raw 2D image data which incorporates in an integrated manner all processing stages—feature extraction, feature association (hard or “soft” matching), nonlinear state estimation.

This paper is focused on the feature extraction. Considered is a feature-based rigid object representation which is more complete as compared to previous techniques. In this approach the features provide information regarding the location and type of significant object components and connections. A robust enhanced Hough transform approach is proposed for line and junction detection. Based on a perspective

*Research supported by NSF Grant ECS-9734285, and NASA/LEQSF grant (2001-4)-01.

¹i.e., the target is considered as a mass-point.

²e.g., the coordinates of some body features in a body frame.

projection imaging model the 2D descriptive and relational features are used for estimating the 3D motion of a rigid body. The proposed technique is evaluated for various objects and image sequence scenarios.

The rest of the paper is organized as follows. Section 2 provides problem formulation, including the models used. An outline of the overall tracking algorithm is given in Section 3. Section 4 provides a detailed description of the proposed enhanced feature detection via a modified Hough transform. Section 5 presents simulation results of feature extraction, and Section 6 provides conclusion further work directions

2 Problem Formulation

2.1 2D Feature observation model

The image acquisition is modeled by the well known perspective projection imaging model of a pin-hole camera [1, 2, 4, 3]. The object is represented though a set of feature points $P^{(i)}$ of a rigid body with IDs $i = 1, 2, \dots, N$.

Let X and Y denote the axes of the 2D image coordinate system on the image plane of a static camera (Fig. 1). An inertial coordinate system $\mathcal{I} = O_{xyz}$ is defined with respect the camera such that its z -axis is pointing along the optical axis of the camera, and x and y axes are parallel to X and Y , respectively. If $p_k^{(i)} = (x_k^{(i)}, y_k^{(i)}, z_k^{(i)})'$ denotes the position vector of the i th feature point $P^{(i)}$ at time k in \mathcal{I} then the image plane position measurements $X_k^{(j)}$ and $Y_k^{(j)}$ are

$$X_k^{(j)} = \phi \frac{x_k^{(i)}}{z_k^{(i)}} + n_{X_k^{(j)}} \quad (1)$$

$$Y_k^{(j)} = \phi \frac{y_k^{(i)}}{z_k^{(i)}} + n_{Y_k^{(j)}} \quad (2)$$

where $j = 1, 2, \dots, M$ denotes the measurement ID, f is the camera focal length³, and $n_{X_k^{(j)}}, n_{Y_k^{(j)}}$ denote the observation errors. The observation errors are resulting from the imaging process and accuracy of the feature extraction, and are assumed to be random, zero-mean, white Gaussian noise.

Note that the feature-to-measurement IDs mapping $i \mapsto j$ is unknown and is subject to the so called *feature matching* problem, addressed below.

³In the sequel it is assumed that $\phi = 1$. In real applications, however, ϕ must be measured or estimated [2], [7].

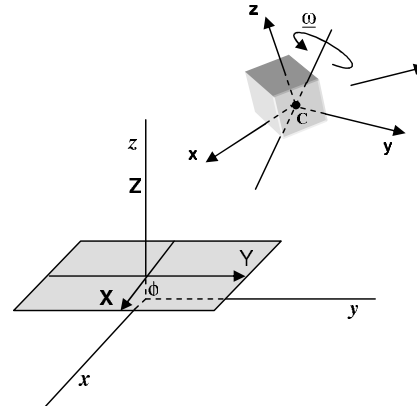


Figure 1: Camera and Object Motion Coordinate Systems

2.2 3D Object structure and motion model

The rigid body object is represented though a set of feature points $P^{(i)}$, $i = 1, 2, \dots, N$. The object structure is defined through the coordinate vector $p^{(i)} = (x^{(i)}, y^{(i)}, z^{(i)})'$ of the feature points $P^{(i)}$ in a coordinate system $\mathcal{C} = C_{xyz}$ which is fixed with the rigid body and $C = C(t)$ is the rotational center of the rigid body motion at time t (Fig. 1). \mathcal{C} is referred to as a *body-frame*⁴. Note that the coordinates $p^{(i)}$ of the feature points in the body frame are time invariant due to the rigidity assumption.

The motion of any feature point $P^{(i)}$ in \mathcal{I} is then given by

$$p^{(i)}(t) = p_c(t) + R(t)p^{(i)} \quad (3)$$

where $p_c = (x_c, y_c, z_c)'$ are the inertial coordinates of vector \overrightarrow{OC} and $R(t)$ is the 3×3 coordinate transformation matrix which aligns \mathcal{C} to \mathcal{I} . $p_c(t)$ describes the motion of the rotation center C (referred to as translational motion) while $R(t)$ describes the rotational motion of the rigid body around C . The rotational matrix $R(t)$ can be conveniently described as an explicit function $R(t) = R(q(t))$ (see [1]) of the unit quaternion $q(t) = (q_1(t), q_2(t), q_3(t), q_4(t))'$ which propagates in time according to the equation

$$\dot{q} = \Omega(\omega(t))q(t), \quad \Omega(\omega) = \frac{1}{2} \begin{bmatrix} 0 & \omega_z & -\omega_y & \omega_x \\ -\omega_z & 0 & \omega_x & \omega_y \\ \omega_y & -\omega_x & 0 & \omega_z \\ -\omega_x & -\omega_y & -\omega_z & 0 \end{bmatrix} \quad (4)$$

⁴The body frame coordinates are given in Sans Serif font in contrast to the inertial coordinates given in Times font.

where $\omega = (\omega_x, \omega_y, \omega_z)'$ is the angular velocity vector

2.3 State-space model

According to (3), (4) an object motion model can be defined for any feature point $P^{(i)}$, $i = 1, 2, \dots, N$ by assuming some type of evolutions of the rotational center $p_c(t)$ and angular velocity $\omega(t)$, i.e. by specifying $\frac{d^n}{dt^n} p_c(t)$ and $\frac{d^n}{dt^n} \omega(t)$. Depending on the targets of interest a great variety of models can be used in different applications – see [14] for a comprehensive survey of maneuvering target models. The multiple model approach [15] is also applicable [16]. Below we adopt the continuous-discrete state space model

$$\dot{\xi} = f(\xi(t)) + w(t), \quad w(t) \sim \mathcal{N}(0, Q(t)) \quad (5)$$

$$\eta_k = h(\xi(kT)) + v_k, \quad v_k \sim \mathcal{N}(0, R_k) \quad (6)$$

obtained through (3), (4) under the assumption of constant translational and angular velocities⁵ and accounting for (1), (2) [1].

The state function f of the target model (5) is given by

$$f(\xi) = f \begin{bmatrix} \xi_1 \\ \xi_2 \\ \xi_3 \\ \xi_4 \\ \xi_5 \\ \xi_6 \\ \xi_7 \\ \xi_8 \\ \xi_9 \\ \xi_{10} \\ \xi_{11} \\ \xi_{12} \\ \xi_{13} \\ \xi_{14} \\ \xi_{15} \\ \vdots \\ \xi_{10+3N} \\ \xi_{11+3N} \\ \xi_{12+3N} \end{bmatrix} = \begin{bmatrix} \xi_3 - \xi_1 \xi_5 \\ \xi_4 - \xi_2 \xi_5 \\ -\xi_3 \xi_5 \\ -\xi_4 \xi_5 \\ -\xi_5^2 \\ \frac{1}{2}(\xi_{12} \xi_7 - \xi_{11} \xi_8 + \xi_{10} \xi_9) \\ \frac{1}{2}(-\xi_{12} \xi_6 + \xi_{10} \xi_8 + \xi_{11} \xi_9) \\ \frac{1}{2}(\xi_{11} \xi_6 - \xi_{10} \xi_7 + \xi_{12} \xi_9) \\ \frac{1}{2}(-\xi_{10} \xi_6 - \xi_{11} \xi_7 + \xi_{12} \xi_8) \\ 0 \\ 0 \\ 0 \\ -\xi_9 \xi_5 \\ -\xi_{10} \xi_5 \\ -\xi_{11} \xi_5 \\ \vdots \\ -\xi_{10+3N} \xi_5 \\ -\xi_{11+3N} \xi_5 \\ -\xi_{12+3N} \xi_5 \end{bmatrix} \quad (7)$$

where

$$\xi \triangleq \left[\frac{x_c}{z_c}, \frac{y_c}{z_c}, \frac{p'_c}{z_c}, q', \omega', \frac{p^{(1)'}}{z_c}, \frac{p^{(2)'}}{z_c}, \dots, \frac{p^{(N)'}}{z_c} \right]'$$

is the normalized state vector. The normalization with $z^c(t)$ is needed to compensate the unknown scale factor inherent to the projection imaging model.

⁵A more precise, multiple model, will be implemented in the further work for a realistic scenario.

The measurement function h of the measurement model (6) is [3]

$$h(\xi) = [h^{(1)}(\xi)', h^{(2)}(\xi)', \dots, h^{(M)}(\xi)']'$$

where

$$h^{(j)}(\xi) = \frac{1}{\gamma_j(\xi)} \begin{bmatrix} \xi_1 + \alpha_j(\xi) \\ \xi_2 + \beta_j(\xi) \end{bmatrix}, \quad j = 1, 2, \dots, M \quad (8)$$

with

$$\begin{aligned} \alpha_j(\xi) &= (\xi_6^2 - \xi_7^2 - \xi_8^2 + \xi_9^2) \xi_{10+3j} + \\ &\quad 2(\xi_6 \xi_7 - \xi_8 \xi_9) \xi_{11+3j} + 2(\xi_6 \xi_8 + \xi_7 \xi_9) \xi_{12+3j} \\ \beta_j(\xi) &= 2(\xi_6 \xi_7 + \xi_8 \xi_9) \xi_{10+3j} + \\ &\quad (-\xi_6^2 + \xi_7^2 - \xi_8^2 + \xi_9^2) \xi_{11+3j} + 2(\xi_7 \xi_8 - \xi_6 \xi_9) \xi_{12+3j} \\ \gamma_j(\xi) &= 1 + 2(\xi_6 \xi_8 - \xi_7 \xi_9) \xi_{10+3j} + \\ &\quad 2(\xi_7 \xi_8 + \xi_6 \xi_9) \xi_{11+3j} + (-\xi_6^2 - \xi_7^2 + \xi_8^2 + \xi_9^2) \xi_{12+3j} \end{aligned}$$

3 Overall tracking algorithm outline

The outline of the overall tracking algorithm is depicted in Fig. 3. It operates in a recursive manner as briefly explained below.

Feature Extraction. When a new frame is acquired the feature extraction (FE) detect line segments, estimates their endpoints, and their intersection (common) points. The set of all feature points consists of all endpoints and intersection (common) points. The features are grouped into subsets (tokens) referred to as junctions. A junction consists of a central and peripheral points. The central point is directly connected to all peripheral points, while a peripheral point is connected directly to the center within a considered junction. See Fig. 2. The number of peripheral points is characteristic for a junction and is called its rank. The representation of the object structure through junctions is advantageous in two main aspects. Firstly, the estimating a junction center by the FE algorithm can be done much more accurate as compared, e.g., to estimating the endpoints and the center of a segment [17]. Secondly, the junction representation greatly facilitates the feature matching by exploiting the information about its rank and structure.

Feature Matching. In principle, matching the observed feature points to the predicted can be done just “pointwise” by using well established techniques for data association in the multiple target tracking of “point” targets, such as joint probability data association (JPDA), multiple hypothesis tracking (MHT)

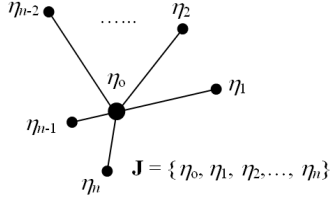


Figure 2: Junction of rank n .

[18], or 2-D assignment algorithms [19]. It is however highly beneficial to perform the association in terms of junctions rather than in terms of single feature points. In this work we implement a simple nearest neighbor (NN) matching logic by using a distance between junctions defined as follows

$$\delta(J, \bar{J}) = \|\eta_0 - \bar{\eta}_0\|^2 + \min_{(i_1, i_2, \dots, i_n)} \sum_{j=1}^n \|\eta_j - \bar{\eta}_{i_j}\|^2 \quad (9)$$

where $J = \{\eta_0; \eta_1, \dots, \eta_n\}$ and $\bar{J} = \{\bar{\eta}_0; \bar{\eta}_1, \dots, \bar{\eta}_n\}$ are a “measured” and a predicted junctions with centers η_0 and $\bar{\eta}_0$, respectively, and $\|\eta - \bar{\eta}\|^2 = (\eta - \bar{\eta})' S^{-1} (\eta - \bar{\eta})$. In the further work we will incorporate this distance to implement the modified auction algorithm for 2-D junction assignment

State Estimation. Provided the feature matching $i \longleftrightarrow j$ is established the state-space model (5), (6) with (7), (8) poses a well defined nonlinear estimation problem. Standard nonlinear estimation techniques, such as extended Kalman filtering (EKF) [7], [3], iterated EKF [1], and particle filtering [9], have (more or less successfully) been used before. In this work we implement the well known unscented Kalman filter [20]. This choice was motivated by the fact that the UKF provides an excellent tradeoff between accuracy and computational load — it has been proven much more accurate than, e.g., the EKF at the expense of a reasonable increase in computation. Moreover, the problem can be of very high dimension and particle filtering may become infeasible for “real-time” implementation.

4 Junction Detection via a Modified Hough Transform

Hough transform is one of the most popular approaches used for line detection [10], [11]. Since any two or more lines having a common point in the image $I(x, y)$ define a junction, line detection may be

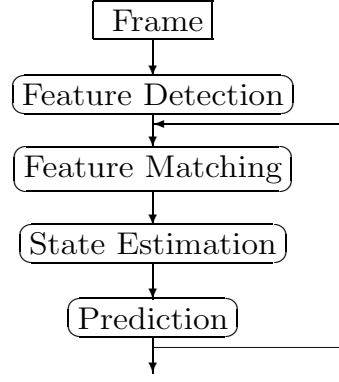


Figure 3: Tracking Algorithm Outline

the basic component of a junction detection procedure. Therefore, in order to detect junctions, a Hough based approach may be employed [12].

Prior to Hough, edge detection is performed resulting in N edge pixels with coordinates (x_j, y_j) , $j = 1, \dots, N$. Hough uses the following parametric equation of a line:

$$\rho = x_j \cos \theta + y_j \sin \theta \quad (10)$$

to build a two-dimensional histogram $A[\rho^{(q)}, \theta^{(q)}]$ of the parameters $\rho^{(q)}$ and $\theta^{(q)}$, which are quantized versions of ρ and θ respectively. More specifically, the angle parameter θ varies assuming quantized values $\theta^{(q)}$ in the interval $[0, 180^\circ]$ to obtain the corresponding ρ parameter. Parameter ρ is also quantized to $\rho^{(q)}$. In order to generate the histogram $A[\rho^{(q)}, \theta^{(q)}]$, this procedure is repeated for every edge pixel. For each pair $(\rho^{(q)}, \theta^{(q)})$, the corresponding histogram bin is incremented:

$$A[\rho^{(q)}, \theta^{(q)}] = A[\rho^{(q)}, \theta^{(q)}] + 1 \quad (11)$$

If $\rho^{(q)} = \rho^{(l)}, \theta^{(q)} = \theta^{(l)}$ correspond to a local maximum in $A[\rho^{(q)}, \theta^{(q)}]$, equation $\rho^{(l)} = x \cos \theta^{(l)} + y \sin \theta^{(l)}$ identifies a line in the image $I(x, y)$. A local maximum point in $A[\rho^{(q)}, \theta^{(q)}]$ is defined as:

$$(\rho^{(l)}, \theta^{(l)}) : A[\rho^{(l)}, \theta^{(l)}] > A[\rho^{(q)}, \theta^{(q)}], \forall (\rho^{(q)}, \theta^{(q)}) \in \mathfrak{N}(\rho^{(l)}, \theta^{(l)}) \quad (12)$$

where $\mathfrak{N}(\rho^{(l)}, \theta^{(l)})$ is a pixel neighborhood around point $(\rho^{(l)}, \theta^{(l)})$.

Although the application of Hough transform is straightforward, its line detection success depends on the quality of the image. Noisy images, and multiple or thick edges may result in false local maximum points, while short line segments may not produce local maxima in $A[\rho^{(q)}, \theta^{(q)}]$.

In this work, a robust Hough-based approach is proposed to identify only true line segments in images. In general, it is expected that the overall maximum in $A[\rho^{(q)}, \theta^{(q)}]$ found at $\rho^{(q)} = \rho^{(m)}, \theta^{(q)} = \theta^{(m)}$, is more probable to correspond to a true line in the image space than other local maxima. Line $l^{(m)} : \rho^{(m)} = x \cos \theta^{(m)} + y \sin \theta^{(m)}$ is the one that corresponds to $A[\rho^{(m)}, \theta^{(m)}]$. Define all edge pixels with a distance less than $d^{(m)}$ from line $l^{(m)}$ as $p_j^{(m)}$, $j = 1, \dots, M$. These M edge pixels are considered as being part of line $l^{(m)}$. The next step is to remove the contribution of these points from the histogram $A[\rho^{(q)}, \theta^{(q)}]$ in order to eliminate their false contribution to other lines. Practically, not all $p_j^{(m)}$ pixels are considered part of a true line segment. Several of those edge pixels may be due to noise, or simply parts of other edge components that happen to be located around line $l^{(m)}$. The actual line component that defines part of a junction should in general be a continuous line segment. Therefore, continuous line segments should be identified from the set of pixels $p_j^{(m)}$. This can be achieved by applying a two-dimensional rotation transformation on the coordinates of $p_j^{(m)}$, namely, $\mathbf{x}^{(m)} = [x_j^{(m)}, y_j^{(m)}]^T$, using $-\theta^{(m)}$ as the angle of rotation. The new coordinates obtained are:

$$\mathbf{x}^{(r)} = \mathbf{R}(-\theta^{(m)})(\mathbf{x}^{(m)})^T \quad (13)$$

where $\mathbf{R}(-\theta^{(q)})$ is the rotation matrix defined as

$$\mathbf{R}(\theta) = \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix} \quad (14)$$

This transformation rotates line $l^{(m)}$ into a horizontal line. Therefore, the line's continuity can be determined by simply examining the continuity of the horizontal $x_j^{(r)}$ coordinates. A threshold T_{sep} can be used to specify if the separation between neighboring $x_j^{(r)}$ coordinates is significant. More specifically, considering the j_o -th rotated point with horizontal coordinate $x_{j_o}^{(r)}$, a significant separation is defined by the following:

$$x_j^{(r)} \notin (x_{j_o}^{(r)}, x_{j_o}^{(r)} + T_{sep}), \forall j \neq j_o \quad (15)$$

Moreover, a threshold T_{long} can be used to specify if a segment is long enough to be considered a true line segment. Considering that $x_{left}^{(r)}$ and $x_{right}^{(r)}$ are, respectively, the segment's leftmost and rightmost points, the segment's length is considered significant if the following holds:

$$x_{right}^{(r)} - x_{left}^{(r)} > T_{long} \quad (16)$$

If multiple line segments are found, they can be considered as different components. However, the contribution of the pixels located on all these segments should be eliminated from histogram $A[\rho^{(q)}, \theta^{(q)}]$. For this purpose, equation (10) is used to find the set of pairs $(\rho^{(q)}, \theta^{(q)})$ that correspond to the line segment pixels similarly to the original Hough. Then, the histogram is updated as:

$$A[\rho^{(q)}, \theta^{(q)}] = A[\rho^{(q)}, \theta^{(q)}] - 1 \quad (17)$$

Nevertheless, the extreme segment points may be excluded, since it is possible that they may belong to different line segments of the same junction. Eliminating them may reduce the chances of identifying those other segments. This procedure is repeated until all significant, based on threshold T_{long} , lines are found. Detected lines with similar $\rho^{(q)}, \theta^{(q)}$ parameters may be considered as a single line. A summary of the algorithmic steps is presented next:

1. Identify significant edges in the image.
2. Apply the original Hough to generate histogram $A[\rho^{(q)}, \theta^{(q)}]$.
3. Identify the maximum peak in $A[\rho^{(q)}, \theta^{(q)}]$. This corresponds to line $l^{(m)}$ in the image space.
4. Draw line $l^{(m)}$ in the image and identify all pixels $p_j^{(m)}$ with a distance $d^{(m)}$ or less from the line.
5. Apply the two-dimensional rotation matrix on the coordinates of points $p_j^{(m)}$, namely $\mathbf{x}^{(m)}$, to obtain new points with coordinates $\mathbf{x}^{(r)}$ located around a horizontal line as in equation (13).
6. Examine the continuity of the horizontal coordinates of the new points $\mathbf{x}^{(r)}$ from left to right. An interval larger than or equal to T_{sep} in which there is no point $p_j^{(m)}$ with corresponding coordinate $x_j^{(r)}$ defines a separation between two line segments. This is described in equation (15).
7. Examine the length of a line segment as in equation (16). If it is less than a threshold T_{long} this line segment is considered too short to be a true line segment.
8. Remove the contribution of all pixels located on the detected line segments from the Hough histogram $A[\rho^{(q)}, \theta^{(q)}]$ as in equation (17).
9. If the maximum peak in $A[\rho^{(q)}, \theta^{(q)}]$ is still more than a threshold, go back to step 4. Otherwise, stop.

It can be easily concluded that the number of operations required for the proposed Hough transform is approximately double compared to the original Hough. Existing techniques used for speeding up the original Hough transform can be also used here. However, this is beyond the scope of this work. Simulation results for the proposed Hough-based technique are presented in the next section.

5 Simulation Results

In this section, simulation results are presented to illustrate the effectiveness of the modified Hough transform technique presented in the last section. The method is evaluated on the simple example of a moving cube. Several states of the cube, including various rotations, are shown in Fig. 4.

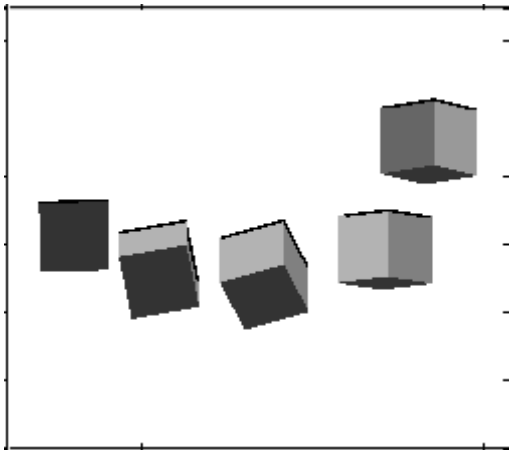


Figure 4: Several stages of a moving cube

Fig. 5 shows one of the cube states after edge detection. The pixel coordinates found on those edges will contribute in the generation of the histogram $A[\rho^{(q)}, \theta^{(q)}]$. Fig. 6 presents the line segments detected for the cube states shown in Fig. 4 using the proposed Hough technique. The pixels at which line segments meet define the junction positions. The current example, although seemingly simple, can cause help in identifying some problems of the traditional Hough transform technique, due to the relatively low image resolution, edge thickness, and parametric similarity of some line segments. For instance the Fig. 4's cube position presented as second from the right, shows a tilted lower face. The two lowest edges of that face could be erroneously considered as one line segment. Increasing the Hough histogram resolution would possibly help in reducing this types of errors.

However, there would be a higher possibility of causing more errors due to lack of sufficient points per histogram bin, resulting in multiple local maxima, and thus unreliable histograms. For instance, Fig. 7 presents the histogram $A[\rho^{(q)}, \theta^{(q)}]$ for one of the cube states. It can be observed that multiple local maxima exist, which make the selection of significant lines considerably difficult. However, the proposed method deals with one line at a time. Fig. 8 depicts the histogram after two lines are detected and the contribution of the corresponding image pixels is removed. Therefore, the third significant line in this histogram is clearer compared to the original histogram of Fig. 7.

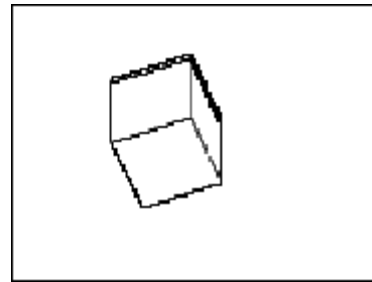


Figure 5: Cube after edge detection

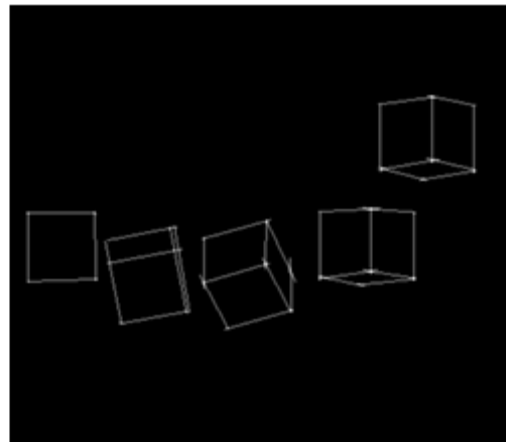


Figure 6: Line segments found for the several cube examples of Fig. 4

Another example of line segment detection is shown in Fig. 9. In this example, one of the line segments is shorter compared to rest, and it's presence is not obvious in the traditional Hough histogram. On the

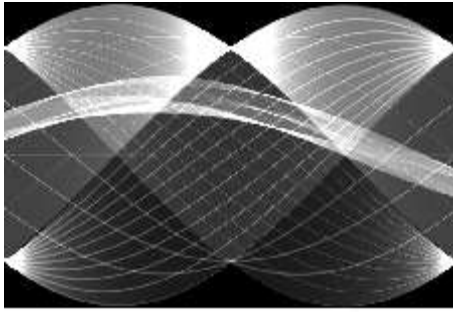


Figure 7: Histogram $A[\rho^{(q)}, \theta^{(q)}]$ for the edges of the cube shown in Fig. 5



Figure 8: Histogram $A[\rho^{(q)}, \theta^{(q)}]$ for the edges of the cube shown in Fig. 5 after two line segments have been removed

other hand, once the pixels corresponding to significant lines are removed, the line segment's presence is apparent.

6 Conclusion and Further Work

The problem of estimating the 3D motion and structure of a rigid object from a sequence of 2D monocular images has been studied. In contrast to most of the available algorithms, addressing only some parts of the problem, an overall feature-based algorithm has been outlined and important issues has been discussed.

The focus of this paper has been on the feature extraction part of the overall tracking algorithm. A new enhanced robust Hough transform algorithm has been proposed for detection of lines and junctions. The advantage of the proposed technique compared to the original Hough is that - similarly to cascade Hough transforms - it operates in both image and Hough domains. However, in contrast to cascade Hough

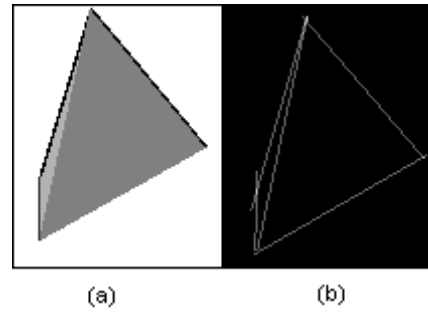


Figure 9: (a) Pyramid-shaped object, and (b) extracted line segments

transforms the proposed technique modifies the image space directly by eliminating line contributions, one line at a time. Thus, the proposed algorithm is capable of identifying short line segments, as well as distinguishing between line segments defined by similar line parameters. The simulation results have demonstrated the above enhancements.

The future work includes incorporation of the new HT algorithm within the overall tracking algorithm including an auction-based assignment algorithm for feature matching and unscented Kalman filter for nonlinear estimation.

References

- [1] T. J. Broida, S. Chandrashekar, and R. Chellappa. Recursive 3-D Motion Estimation from a Monocular Image Sequence. *IEEE Trans. Aerospace and Electronic Systems*, 26(3):639 – 656, Jul. 1990.
- [2] T. J. Broida and R. Chellappa. Estimating the Kinematics and Structure of a Rigid Object from a Sequence of Monocular Images. *IEEE Trans. Patt. Anal. and Mach. Intellig.*, 13(6):497 – 513, Jun. 1991.
- [3] S. D. Blostein, L. Zhao, and R. M. Chann. Three-Dimensional Trajectory Estimation from Image Position and Velocity. *IEEE Trans. Aerospace and Electronic Systems*, 36(4):1075 – 1089, Oct. 2000.
- [4] G. S. Young, R. Chellappa, and T. H. Wu. Monocular Motion Estimation Using a Long Sequence of Noisy Images. In *Proc. IEEE International Conf. on Acoustics, Speech and Signal Processing, ICASSP-91*, pages 2437 – 2440, 14-17 Apr. 1991.

- [5] J. Weng, N. Ahuja, and T. S. Huang. Optimal Motion and Structure Estimation. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 15(9):864 – 884, Sep. 1993.
- [6] N. Cui, J. Weng, and P. Cohen. Extended Structure and Motion Analysis from Monocular Image Sequences. In *Proc. Third International Conference on Computer Vision.*, pages 222 – 229, 4-7 Dec. 1990.
- [7] A. Azarbayejani and A. Pentland. Recursive Estimation of Motion, Structure, and Focal Length. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 17(6):562 – 575, 1995.
- [8] T. Jebara, A. Azarbayejani, and A. Pentland. 3D Structure from 2D Motion. *IEEE Signal Processing Magazine*, 6(3):66 – 84, 1999.
- [9] G. Qian, R. Chellappa, and Q. Zheng. Bayesian Algorithms for Simultaneous Structure From Motion Estimation of Multiple Independently Moving Objects. *IEEE Trans. on Image Processing*, 14(1):94– 109, Jan. 2005.
- [10] E. R. Davies. Application of the Generalised Hough Transform to Corner Detection. *IEE Proc. E*, 135(1):49 – 54, Jan. 1988.
- [11] T. Tuytelaars, M. Proesmans, and L. Van Gool. The Cascaded Hough Transform. In *Proc. International Conf. On Image Processing*, pages 736 – 739, Oct. 1997.
- [12] W. A. Barrett and K. D. Petersen. Houghing the Hough: Peak Collection for Detection of Corners, Junctions and Line Intersections. In *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages II-302 – II-309, Dec. 2001.
- [13] T. S. Huang and A. N. Netravali. Motion and Structure from Feature Correspondances: A Review. *Proc. IEEE*, 82(2):252–268, 1994.
- [14] X. R. Li and V. P. Jilkov. Survey of Maneuvering Target Tracking—Part I: Dynamic Models. *IEEE Trans. Aerospace and Electronic Systems*, AES-39(4):1333–1364, Oct. 2003.
- [15] X. R. Li and V. P. Jilkov. A Survey of Maneuvering Target Tracking—Part V: Multiple-Model Methods. *IEEE Trans. Aerospace and Electronic Systems*, 41(2), 2005. To appear.
- [16] K. J. Bradshaw, I. D. Reid, and D. W. Murray. The Active Recovery of 3D Motion Trajectories and their Use in Prediction. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 19(3):219–234, Mar. 1997.
- [17] Z. Zhang and O. D. Faugeras. Three-Dimensional Motion Computation and Object Segmentation in a Long Sequence of Stereo Frames. Technical Report 1438, INRIA, France, Jul. 1991.
- [18] Y. Bar-Shalom and X. R. Li. *Multitarget-Multisensor Tracking: Principles and Techniques*. YBS Publishing, Storrs, CT, 1995.
- [19] K. Pattipati, R. L. Popp, and T. Kirubarajan. Survey of Assignment Techniques for Multitarget Tracking. In Y. Bar-Shalom and W. D. Blair, editors, *Multitarget-Multisensor Tracking: Applications and Advances, Vol. III*, Dedham, MA, 2000. Artech House.
- [20] S. J. Julier and J. K. Uhlmann. Unscented Filtering and Nonlinear Estimation. *Proc. IEEE*, 92(3):401–422, March 2004.