

04295
144

Detection of Obstacles on Runway Using Ego-Motion Compensation and Tracking of Significant Features

**An Interim Technical Report for NASA Ames Cooperative Agreement No. NCC2-916
"A Vision-Based Obstacle Detection System for Aircraft Navigation"
Period of the Grant: August 1, 1995 to July 31, 1997**

**Submitted to
NASA Ames Research Center
Technical Officer: Phillip N. Smith
Flight Deck Branch, 210-1
Moffett Field, CA 94035-1000**

by

**Rangachar Kasturi and Octavia Camps
Principal Investigators
Department of Computer Science and Engineering
The Pennsylvania State university
University Park, Pennsylvania 16802**

**Tel: (814) 863-4254 Fax: (814) 865-3176
E-Mail: {kasturi,camps}@cse.psu.edu**

**Graduate Students:
Tarak Gandhi
Sadashiva Devadiga**

June 14, 1996

Detection of Obstacles on Runway using Ego-Motion Compensation and Tracking of Significant Features

Abstract

This report describes a method for obstacle detection on a runway for autonomous navigation and landing of an aircraft. Detection is done in presence of extraneous features such as tire-marks. Suitable features are extracted from the image and warping using approximately known camera and plane parameters is performed in order to compensate ego-motion as far as possible. Residual disparity after warping is estimated using an optical flow algorithm. Features are tracked from frame to frame so as to obtain more reliable estimates of their motion. Corrections are made to motion parameters with the residual disparities using a robust method, and features having large residual disparities are signaled as obstacles. Sensitivity analysis of the procedure is also studied. Nelson's optical flow constraint is proposed to separate moving obstacles from stationary ones. A Bayesian framework is used at every stage so that the confidence in the estimates can be determined.

1 Introduction

The goal of this research is to detect stationary and/or moving obstacles on a runway for autonomous navigation and landing of an aircraft. The existing methods for this task can be classified as either feature based or optical flow based.

In the feature based methods, significant features are detected in the images and matched from one frame to another. An example of this approach is the method proposed by Sridhar et al. [9, 10] to detect and track stationary obstacles on a runway. In this method, features are matched in adjacent frames using normalized correlation. A Kalman filter is then used to track the features in subsequent frames and to produce the range map used to detect the obstacles.

In the flow based methods, an optical flow field is obtained for the entire image. Using the optical flow and the information about the motion of camera, obstacles are detected. For example, in the method proposed by Sull et al. [11, 12], the runway is modeled as a planar surface and an initial model flow field is computed using the data from the inertial navigation unit (INU). Two image frames are warped from one to the other using the given motion and plane parameters. This warping compensates the ego-motion of the features that are stationary and are located on the runway plane, as far as the accuracy of these parameters permit. The residual optical flow is estimated and places where this is greater than a threshold are considered for obstacle detection. The residual optical flow at the remaining places is assumed to be due to inaccuracy of the model flow field, and is used to improve the model accuracy. Using the new model, warping is redone, and places where there is significant disparity are signaled as obstacles. This procedure is capable of detecting stationary obstacles located at a height above the runway plane, or moving obstacles. Extraneous features such as tire-marks are also separated.

In this report, we present a new method that combines the advantages of the feature based and the flow based methods. The proposed method resembles Sull's approach [11, 12] in that warping is used to compensate the ego-motion of the camera. However, instead of finding optical flow over the entire image, features are selected from the image and the optical

Contents

1	Introduction	1
2	Imaging Geometry	2
2.1	Coordinate Systems	2
2.2	Coordinate Transformation	4
2.3	Perspective Projection Equations	5
3	System Overview	5
4	Feature Detection	8
5	Warping	10
6	Determination of Optical Flow	14
7	Tracking	17
8	Obstacle Detection	19
8.1	Sensitivity Analysis	20
8.2	Improvement of plane parameters	24
9	Separating Moving Objects	26
10	Observations and results	29
11	Summary and Future Work	34

flow is estimated only at these features. This approach results in speed up of computation and ease of systematic tracking of features. Furthermore, a statistical framework is used at every step to obtain not only reliable estimates of parameters, but also an estimate of their covariance.

2 Imaging Geometry

The imaging system is depicted in Fig. 1, where the aircraft is equipped with the Inertial Navigation System (INS), which in cooperation with the ground equipment, constantly provides the position and orientation of the aircraft. A camera is mounted in front of the aircraft to capture the images. Also shown in the figure are the relationships among the different coordinate systems which are described below.

2.1 Coordinate Systems

As shown in Fig. 1, the coordinate systems of the vision system include the earth frame, the aircraft body frame, the sensor frame, the image plane axes and the pixel axes [6].

1. Earth (World) frame: The earth frame is rigidly affixed to the earth with three axes, X_e , Y_e and Z_e . The origin of the earth frame is an arbitrarily selected point on the runway.
2. Aircraft body frame: The aircraft body axis frame (or body frame) is assumed to be fixed relative to the aircraft with the X_b axis pointing forward out the aircraft nose, the Y_b axis pointing out the right hand side of the aircraft, and the Z_b axis pointing downward relative to the aircraft's geometry. The origin of the body frame is the aircraft's nominal center of gravity.
3. Sensor frame: The sensor frame is rigidly attached to the camera and originates at the lens focal point. The Y_s and Z_s axes are parallel to the image plane axes u and v .

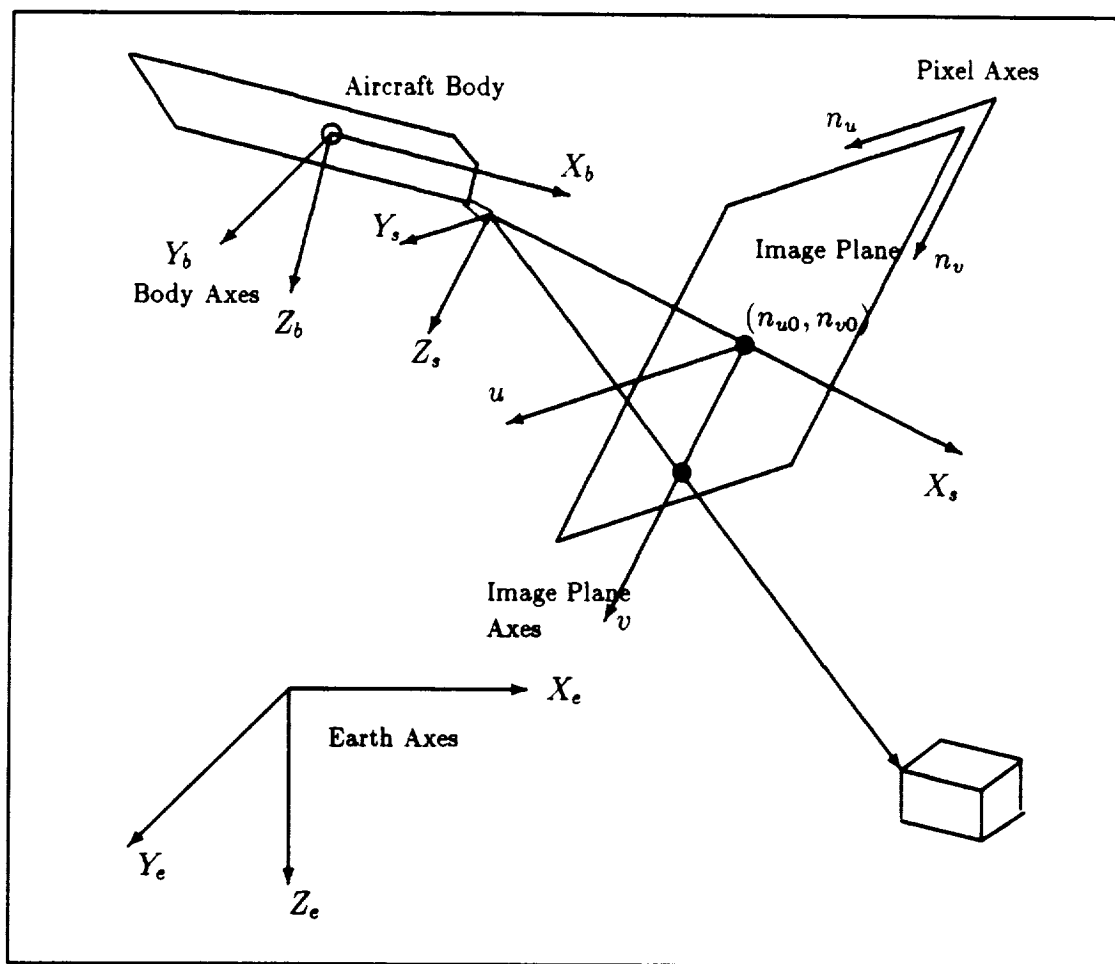


Figure 1: Geometry for the imaging system

respectively. The X_s axis points along the optical axis. The camera is rigidly mounted on the aircraft.

4. Image Plane Axis The image plane axes are oriented along the rows and the columns of the sensor array. The u axis points to the right along the rows and the v axis points downward along the columns.
5. Pixel Axes: The pixel axes, n_u and n_v are attached to the camera's image plane and point along the rows and columns of the sensor array as do the image plane axes; however, the pixel axes originate at the upper left-hand corner of the sensor array rather than at the image center. The upper left-hand pixel has the coordinate (0,0).

2.2 Coordinate Transformation

The coordinate transformation from one frame to another can be expressed by:

$$\begin{pmatrix} X' \\ Y' \\ Z' \end{pmatrix} = R(\psi, \theta, \phi) \begin{pmatrix} X - X_t \\ Y - Y_t \\ Z - Z_t \end{pmatrix} \quad (1)$$

or equivalently:

$$r' = R(r - r_t) \quad (2)$$

where $r_t = (X_t, Y_t, Z_t)^T$ is the translation vector in the earth coordinate system, and R is the rotation matrix. The rotation matrix is defined in terms of Euler angles of yaw (ψ),

Rotation matrices between earth, body and sensor frames are related by:

$$R_{se} = R_{sb}R_{be} \quad (4)$$

where R_{se} , R_{sb} , R_{be} denote the rotation matrices between sensor and earth, sensor and body, and body and earth, respectively.

Equivalently, the transformation can also be expressed as:

$$r' = Rr + s \quad (5)$$

where $s = (s_1, s_2, s_3)$ is the translation vector in the camera coordinate system.

2.3 Perspective Projection Equations

The perspective projection of a point on the image plane is shown in Fig. 2. Using simple geometry, the perspective projection equations mapping points from the sensor axis system to the image plane system are:

$$u = f \frac{y_s}{x_s}, v = f \frac{z_s}{x_s} \quad (6)$$

where (x_s, y_s, z_s) is the location of a point in the sensor axes, (u, v) is its projected location on the image plane and f is the effective focal length of the camera.

The pixel coordinates in the image array are given by:

$$n_u = n_{u0} + Au, n_v = n_{v0} + v \quad (7)$$

where (n_{u0}, n_{v0}) are the actual pixel coordinates of the image center, and A is the aspect ratio defined as $\Delta v/\Delta u$ where Δu and Δv are the horizontal and vertical pixel spacing, respectively.

3 System Overview

The input to the system is a sequence of images captured from the camera onboard, position and velocity (both linear and angular) of the aircraft obtained from sources such as the

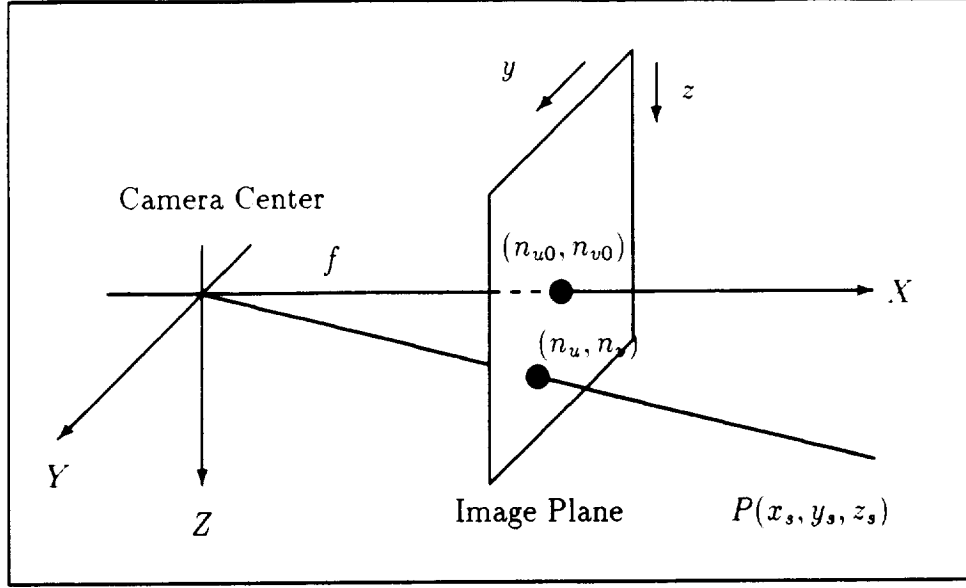


Figure 2: Perspective projection

GPS and INS, referred here as the Inertial Navigation Unit (INU), and knowledge about the parameters of the runway plane. Using these parameters, a transformation which maps features from one frame to another, known as warping, is obtained. Significant features are detected in the image and warping is applied to these. Using the warped positions of the features, optical flow showing the motion of features from frame to frame is obtained. Due to warping, the ego-motion of the features on the runway is compensated as much as possible and residual flow is obtained. This makes it easier to detect independently moving obstacles as well as the obstacles with height. Once the optical flow is obtained, features are tracked from one frame to the other and velocities of the features are smoothed using moving average filtering. The estimated residual velocity can then be thresholded in order to check which features are moving or are at a height above the runway plane. Residual velocities of the remaining features can be used to correct the inaccuracies of the warping parameters. The system block diagram is shown in Fig. 3.

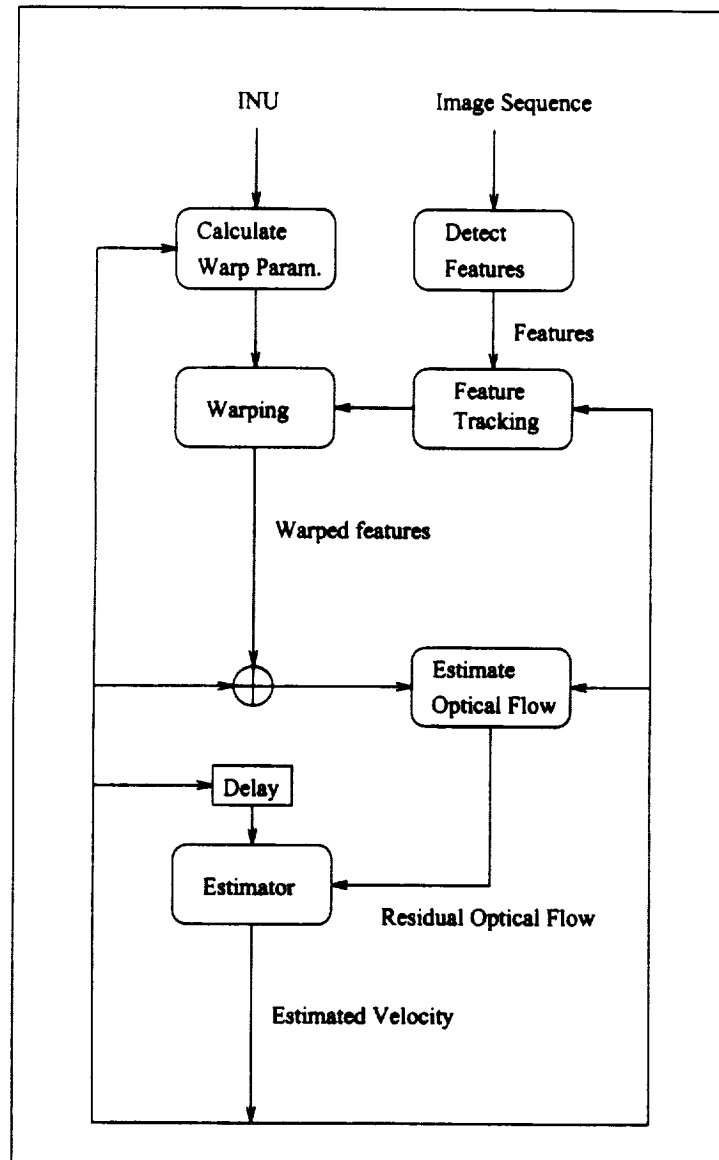


Figure 3: System block diagram.

4 Feature Detection

Optical flow is defined as the apparent movement of a point on the image in the time duration of one frame. In order to estimate the optical flow, there must be a significant intensity variation. Consider for example, the image of a rectangular box shown in Fig. 4(a). Fig. 4(b) shows a background area of the image zoomed in. Movement in any direction has no effect on this part of the image since there is no variation of intensity. Thus, motion cannot be detected from areas of the image having no significant intensity variation. Now consider the portion of the image zoomed in Fig. 4(c), showing only an edge of the box. If the edge translates in the direction perpendicular to itself, the image changes, and the motion can be detected. However, assuming an infinite edge, if it moves in the direction parallel to itself, the image does not change and the motion cannot be detected. This shows, that on an edge, one can detect only normal motion, i.e. motion perpendicular to the direction of the edge. This effect is known as aperture effect since it is like observing a long edge through a small aperture of a camera. On the other hand, consider the portion of the image showing a corner of the box as in Fig. 4(d). Movement in any direction would produce change in the image. Hence, full motion can be detected at corners in the image.

As seen from the above argument, optical flow cannot be determined in uniform (low variance) areas of the image. Also, on edges, one can determine only the normal optical flow. But on corners, full optical flow can be determined. Based on these arguments, corners are good candidates to use as features to track using the estimated optical flow.

Edges are characterized by a differential gradient in one particular direction. By applying a discrete differential operator, one can obtain the edges in the image. Strictly speaking, the gradient is not defined at corner points. However, in the neighborhood of the corner, gradient in at least two directions is observed at some points. This fact can be used to obtain corners in the image.

Differentiation is a noisy operator and hence, in real images contaminated with noise, one has to smooth the image before applying a differential operator. But this blurs the edges and corners of the image. Smith and Brady [7, 8] have used a different approach known as

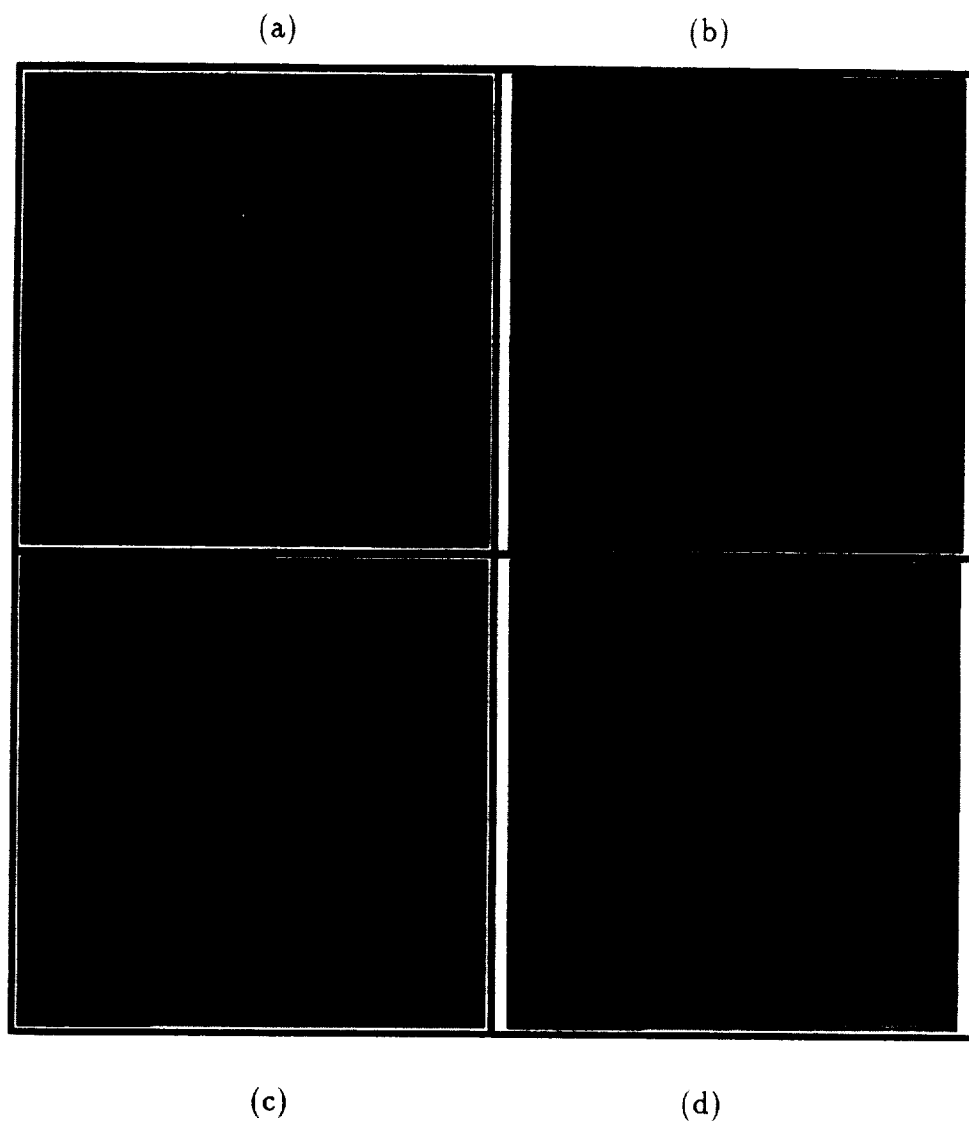


Figure 4: Potential of motion detection: (a) Rectangular box image (b) Background portion of the image: Motion cannot be detected (c) An edge of the rectangle: Only motion perpendicular to the edge can be detected (d) A corner of the rectangle: Full motion can be detected

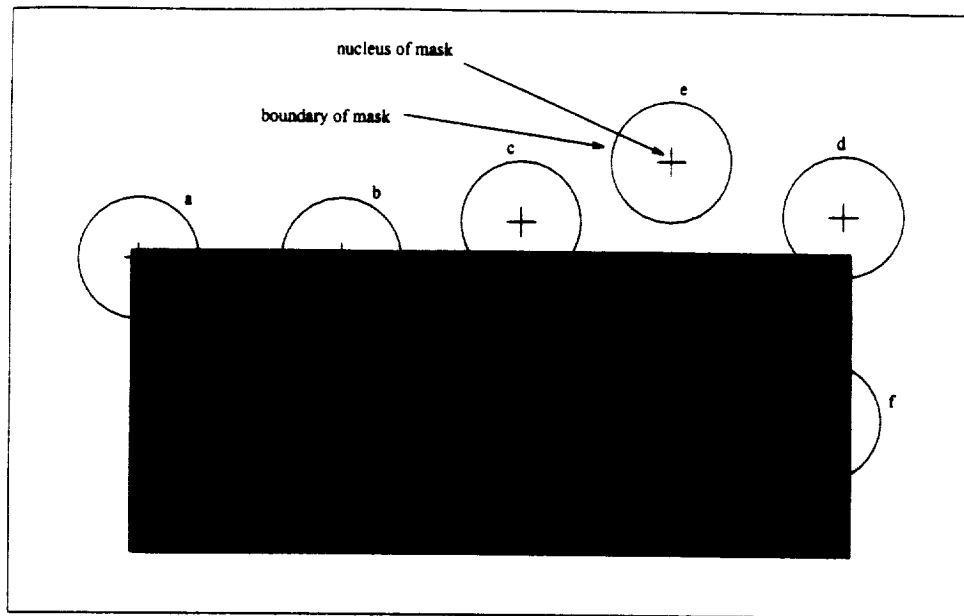
the SUSAN principle to locate edges and corners in an image.

Consider a rectangular box as shown in Fig. 5(a). A circular mask is shown at five different positions in the image. The central pixel of the mask is known as the nucleus. The area of the mask having the same (or similar) intensity as the intensity of the nucleus is known as “USAN” i.e. Univalve Segment Assimilating Nucleus. In Fig. 5(b), the USAN is shown in white color. It can be observed that the USAN area is largest in the uniform portions of the image, smaller in the edge areas and smallest near the corners of the image. Using this principle, one can locate the edges and corners of the image by taking local minima of USAN. Since the minimum is taken, the principle is known as SUSAN or Smallest Univalve Segment Assimilating Nucleus.

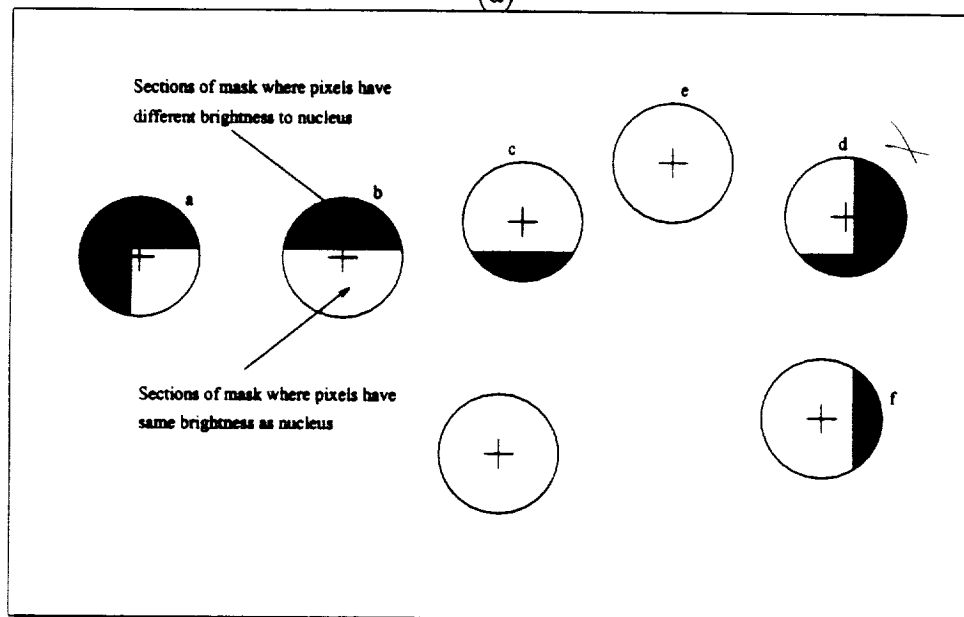
The SUSAN corner detector was applied to video image sequences captured from a helicopter, and supplied by NASA. In Fig. 6, frame number 50 of the image sequence *runway_crossing_new* of a truck crossing the runway is shown. Fig. 7(a) shows a zoomed part of the image and the output of the corner detector is shown in Fig. 7(b). The detector detects corner-like features where there is intensity variation in at least two directions. It can be seen that not only the features corresponding to the moving truck, but also extraneous features such as tire-marks are also detected by the corner detector.

5 Warping

The features found using the SUSAN corner detection algorithm are used to detect obstacles. However, due to the motion of the camera, even extraneous features undergo movement from frame to frame. Furthermore, if the range of a feature is unknown, the motion that it undergoes cannot be uniquely defined. Hence, features are first assumed to be stationary and on the runway plane, so that their range is unique. The corresponding position of the feature in another frame can then be calculated using the information about the motion of the camera and the equation of the runway plane. Moving obstacles and obstacles at height do not satisfy this constraint, and therefore will have different disparities from what



(a)



(b)

Figure 5: SUSAN Principle (a) Circular masks at different places on the image (b) USAN shown in white color.

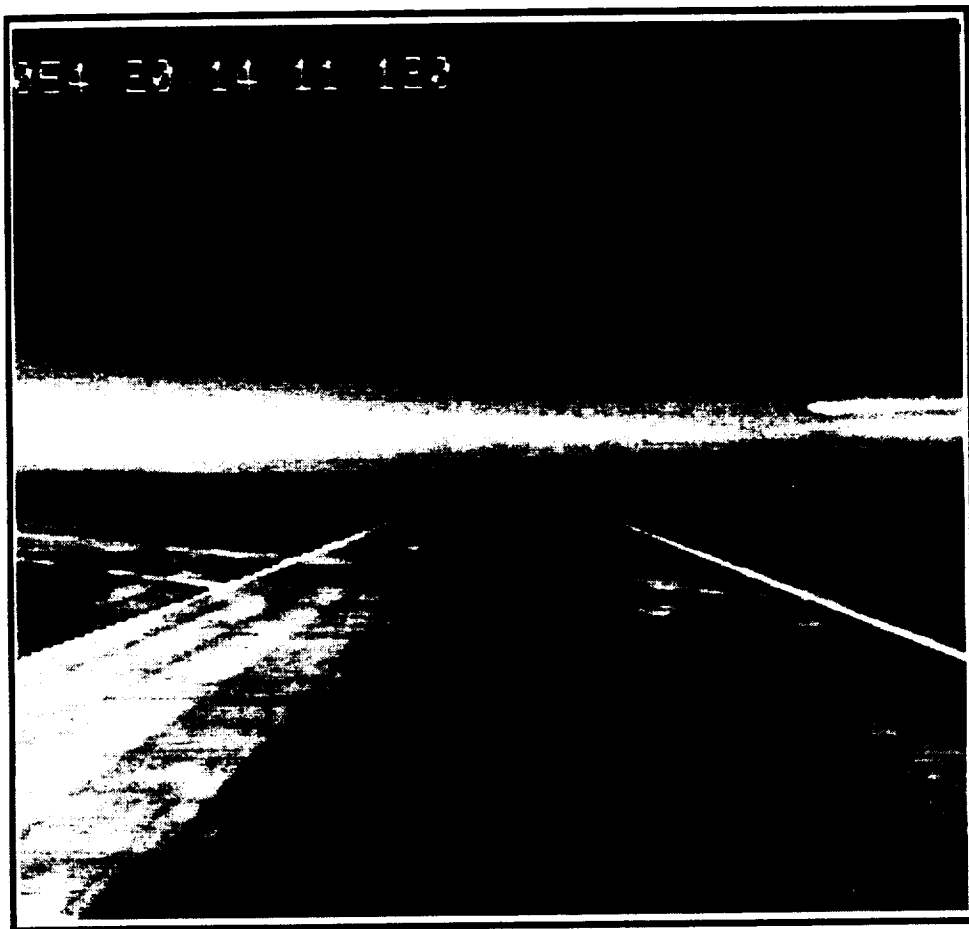


Figure 6: Frame 50 in the image sequence *runway_crossing_new*.

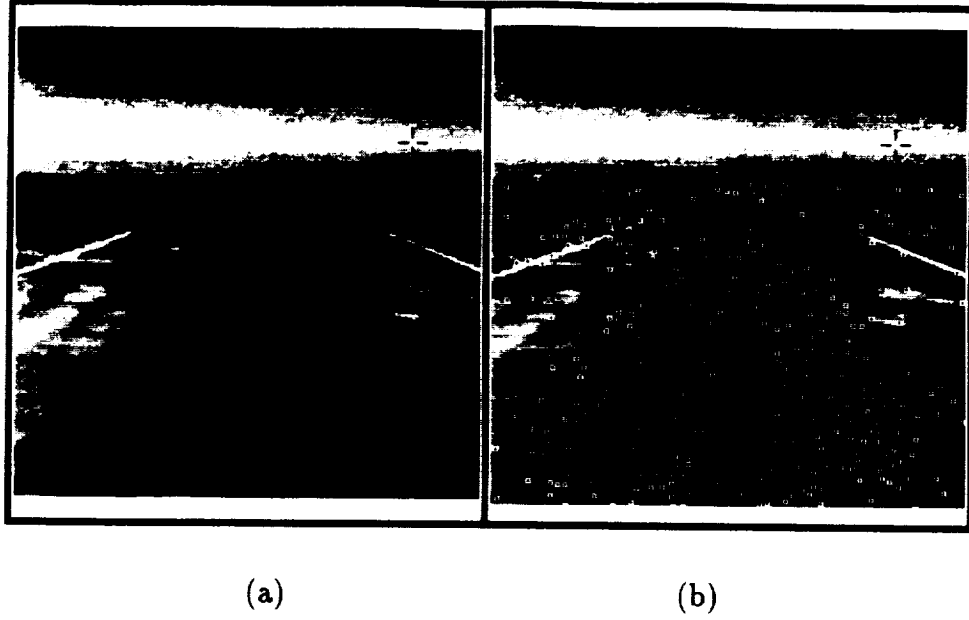


Figure 7: Application of the SUSAN corner detector: (a) Original image (zoomed part)
(b) Detection of corner-like features

is predicted by this method.

Let $r = (x, y, z)^T$ be the coordinates of a point in a camera reference frame. The perspective projection of the point has image coordinates $(u, v)^T$ given by

$$u = \frac{fy}{x}, v = \frac{fz}{x} \quad (8)$$

where f is the focal length of the camera.

Consider that the camera undergoes a rotation R and a translation s so that the coordinates of a point changes from $r = (x, y, z)^T$ in the first frame to $r' = (x', y', z')^T$ in the second frame with

$$r' = Rr + s \quad (9)$$

Let the runway plane be modeled by the equation:

$$n^T r = n_1 x + n_2 y + n_3 z = 1 \quad (10)$$

where $n = (n_1, n_2, n_3)^T$ is the normal vector of the runway plane in the reference frame of the first position. Then, the mapping of the image features on the runway, from the first

frame to the second, is given by the warping matrix:

$$A = R + sn^T \quad (11)$$

Thus, the image coordinates (u, v) of a point on the runway plane are mapped to new image coordinates (u', v') according to the following equations:

$$\begin{aligned} u' &= f \frac{A_{21}f + A_{22}u + A_{23}v}{A_{11}f + A_{12}u + A_{13}v} \\ v' &= f \frac{A_{31}f + A_{32}u + A_{33}v}{A_{11}f + A_{12}u + A_{13}v} \end{aligned} \quad (12)$$

Using these equations, features on the runway plane can be warped from one frame to another.

6 Determination of Optical Flow

Discrimination between obstacles and extraneous features on the runway (like tire-marks) can be done by computing the optical flow of the features.

Barron et al. [1] have implemented and evaluated several optical flow algorithms in the literature. According to their survey, the method by Lucas and Kanade [3] provides estimation to subpixel accuracy and performs most consistently and reliably over all their test images. Due to these reasons, a modification of this method by Simoncelli et al. [5] that uses a statistical model to account for noise is used. This method not only provides the estimates of optical flow, but also their covariance. The algorithm requires at least five frames, and optical flow is computed in the central frame. Time and space gradients are found at positions of interest and these are used in order to estimate the optical flow.

However, in our method, warped locations of features are used to locate them in the frames, so that the ego-motion of the camera is compensated as much as possible, and only the residual disparities are obtained.

Consider a point in frame t at pixel $w = (u, v)$. The image pixel value at this point is denoted by $F(t, u, v)$. Let $w(t) = (u(t), v(t))$ be a feature in frame t . Assume that the

brightness of the point remains constant over a short duration of time. Then,

$$F(u, v, t) = F(u + \Delta u, v + \Delta v, t + \Delta t) \quad (13)$$

By Taylor expansion to first order, for small values of $\Delta u, \Delta v$, and Δt , we have,

$$F(u + \Delta u, v + \Delta v, t + \Delta t) - F(u, v, t) \simeq F_u \Delta u + F_v \Delta v + F_t \Delta t \quad (14)$$

where F_u, F_v, F_t are partial derivatives of F w.r.t. u, v , and t , respectively. Hence, we have:

$$F_u \Delta u + F_v \Delta v + F_t \Delta t = 0 \quad (15)$$

or in vector notation,

$$F_s d + F_t = 0 \quad (16)$$

where $F_s = (F_u, F_v)$ is the space gradient and $d = (d_u, d_v) = (\Delta u, \Delta v)$ is the optical flow or disparity between adjacent frames. This is known as the **gradient constraint equation** [2].

In order to compensate the ego-motion of the features, warping to map features from one frame to another is performed. The warping of the feature $w = (u, v)$ in any frame t' is denoted by

$$w(t') = (u(t'), v(t')) = \text{warp}(w(t)) \quad (17)$$

Also, we use the following notation:

$$\begin{aligned} t' &= t + \Delta t \\ u' &= u(t + \Delta t) + \Delta u \\ v' &= v(t + \Delta t) + \Delta v \end{aligned} \quad (18)$$

The time and space gradients of the feature are found using:

$$\begin{aligned} F_t(t, u, v) &= \sum_{\Delta t, \Delta u, \Delta v} m_t(\Delta t, \Delta u, \Delta v) F(t', u', v') \\ F_u(t, u, v) &= \sum_{\Delta t, \Delta u, \Delta v} m_u(\Delta t, \Delta u, \Delta v) F(t', u', v') \\ F_v(t, u, v) &= \sum_{\Delta t, \Delta u, \Delta v} m_v(\Delta t, \Delta u, \Delta v) F(t', u', v') \end{aligned} \quad (19)$$

where m_t, m_u, m_v are masks to find gradients in the respective directions, and perform smoothing in other directions. These masks are separable in $\Delta t, \Delta u$ and Δv as follows:

$$\begin{aligned} m_t(\Delta t, \Delta u, \Delta v) &= g_t(\Delta t)h_u(\Delta u)h_v(\Delta v) \\ m_u(\Delta t, \Delta u, \Delta v) &= h_t(\Delta t)g_u(\Delta u)h_v(\Delta v) \\ m_v(\Delta t, \Delta u, \Delta v) &= h_t(\Delta t)h_u(\Delta u)g_v(\Delta v) \end{aligned} \quad (20)$$

where $g(\cdot)$ is the gradient mask and $h(\cdot)$ is the smoothing mask. We used the following masks for $g(\cdot)$ and $h(\cdot)$:

$$\begin{aligned} g(\cdot) &= \begin{pmatrix} 1 & -8 & 0 & 8 & -1 \end{pmatrix} \\ h(\cdot) &= \begin{pmatrix} 1 & 1 & 1 \end{pmatrix} \end{aligned} \quad (21)$$

The center value represents the center of the mask.

However, differentiation is an operator that amplifies the image noise. Furthermore, since we have only one independent equation between two unknowns, we cannot determine the optical flow. In fact, this is the same as aperture effect discussed in the previous section, and only the normal flow i.e. flow in the direction of the gradient can be determined using a single point. However, using a point and its neighborhood, more equations are obtained and the full optical flow can be determined.

In order to account for inaccuracies, the following probabilistic model is used to estimate the optical flow. [5]:

$$F_s^T(d - \eta_1) + F_t = \eta_2 \quad (22)$$

where η_1 and η_2 are independent zero mean Gaussian noises with covariances given by $\sigma_1^2 I_2$ and σ_2^2 . The noise η_1 accounts for errors due to the violation of planarity assumptions and the noise η_2 accounts for errors in the derivative computations.

The full optical flow is determined making the assumption that the flow d is locally constant over a small region around the feature. The time and space gradients are then determined in a window around the feature point.

The maximum likelihood estimate of the optical flow is given by its mean μ_d and covariance Σ_d as follows:

$$\begin{aligned}\mu_d &= -\Sigma_d b \\ \Sigma_d &= \left[M + \Sigma_p^{-1} \right]^{-1}\end{aligned}\tag{23}$$

where

$$\begin{aligned}M &= \sum_{u',v'} F_s(u',v') \sigma^{-2}(u',v') F_s(u',v')^T \\ b &= \sum_{u',v'} F_s(u',v') \sigma^{-2}(u',v') F_t(u',v') \\ \sigma^2 &= \sigma_1^2 \|F_s(u',v')\|^2 + \sigma_2^2\end{aligned}\tag{24}$$

and Σ_p is the prior covariance of the flow.

Fig. 8 shows an image and the computed optical flow using five frames around it along with the covariance ellipses. The flow vectors are magnified 25 times whereas the covariance ellipses are magnified 5 times. It is seen that not only the obstacle, but also the extraneous features have significant optical flows even after warping. This is due to both, inaccuracy of motion and plane parameters, as well as the inaccuracy in estimating the flow.

7 Tracking

Once the features are detected in the initial frame, they are tracked over frames. The estimated residual flow is first added to the feature in order to obtain the expected position of the feature in the next frame. Its location is then warped on to the next frame. Hence, if the location of a feature in frame t is $w(t) = (u(t), v(t))^T$, the tracked location is given by:

$$w(t+1) = \text{warp}(w(t) + d(t))\tag{25}$$

However, instead of using the raw values of optical flow, a moving average of the residual flow is taken. At each frame, the computed residual flow is integrated with the moving

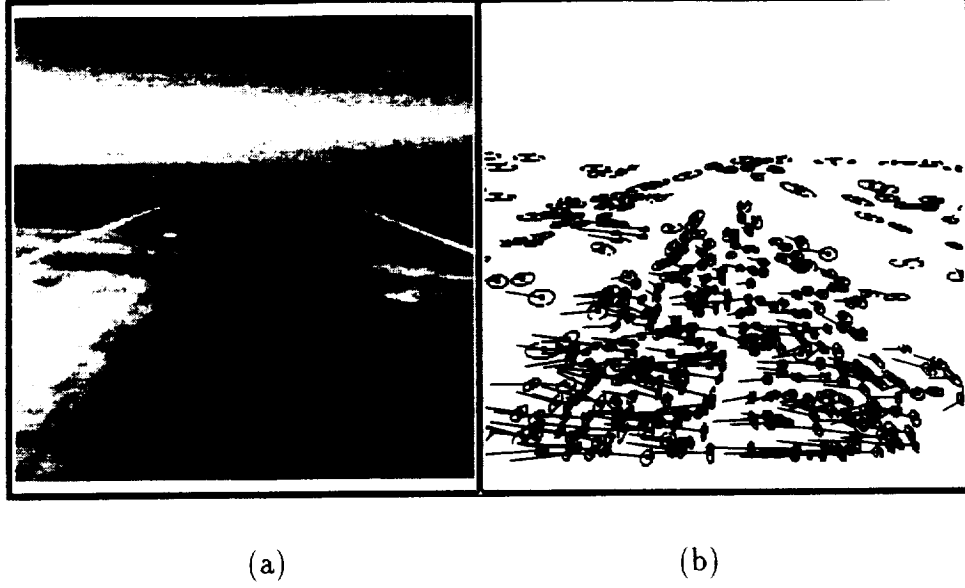


Figure 8: Computation of optical flow (a) Zoomed part of original image (b) Estimates of residual optical flows (unsmoothed) with covariance ellipses.

average. If $d_s(t)$ is the smoothed estimate of flow in frame t and $d(t+1)$ is the raw flow in frame $t+1$, the smoothed flow in frame $t+1$ is given by:

$$\begin{aligned}
 d_s(t+1) &= d_s(t) + k(t)(d(t+1) - d_s(t)) \\
 k(t) &= \frac{1}{\min(t, 5)}
 \end{aligned} \tag{26}$$

In this manner, a smoothed estimate of residual flow is obtained. Instead of this simple moving average method, a Kalman filter can also be used to update the position and velocity of the feature.

Fig. 9 shows the tracked features and the smoothed residual velocity estimates along with the covariance ellipses for a sequence of fifteen images with the image shown in Fig. 6 as the center frame. The flow vectors are magnified 25 times whereas the covariance ellipses are magnified 5 times.

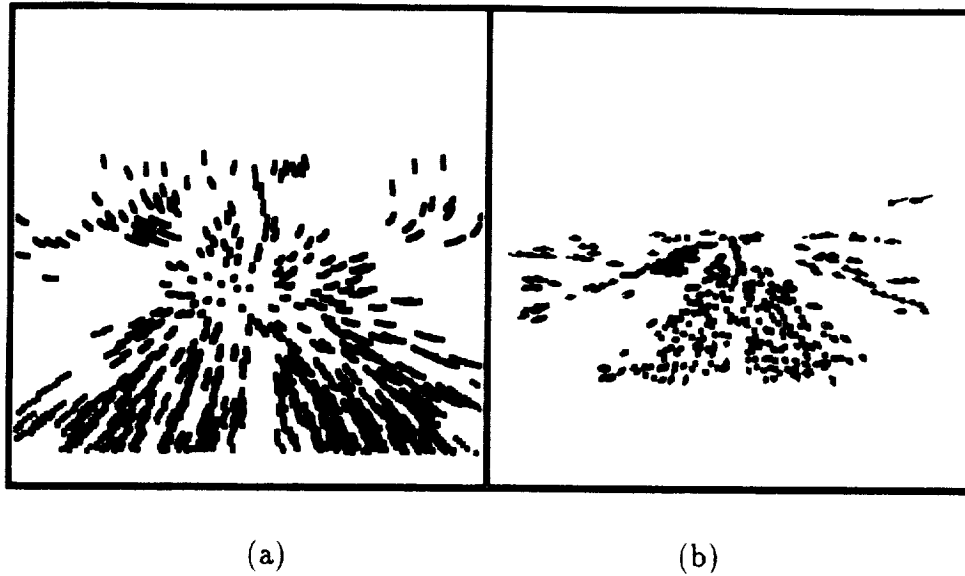


Figure 9: Tracking of features using large number of frames (a) Tracked features in the sequence of images (b) Estimate of residual feature velocities (smoothed)

8 Obstacle Detection

The residual flow nearly compensates the ego-motion for the points on the runway plane and can be used to detect moving obstacles, as well as obstacles which are located above or below the runway plane. However, the accuracy of the parameters used for warping is limited by the accuracy with which the rotation, translation and plane parameters are obtained. Taking this into consideration, two approaches can be taken:

1. Calculate the sensitivity of the warping to the accuracy of the INS and plane parameters. Whenever the residual flow is above the expected deviation, signal an obstacle.
2. Use the residual flow in order to improve the estimate of the warping matrix A . However, a robust method must be used so as to reject the outliers in estimating the parameters.

In order to relate small perturbations in one vector variable to corresponding perturbations in another variable, one can use the Jacobian matrix. Let $y = f(x)$ be an arbitrary vector function of a vector variable x . The Jacobian of y w.r.t. x is denoted here by $J_{y|x}$. Element (i, j) of this matrix is $\partial y_i / \partial x_j$. The chain rule of derivatives can be easily applied to Jacobians using matrix product.

$$J_{z|x} = J_{z|y} J_{y|x} \quad (27)$$

The warping matrix A depends on the camera translation s , rotation R and the plane normal n . Incremental changes in these can be expressed as the linear displacement vector $\Delta s = (\Delta s_1, \Delta s_2, \Delta s_3)^T$, angular displacement vector $\Delta \theta = (\Delta \theta_1, \Delta \theta_2, \Delta \theta_3)^T$, and change in normal vector $\Delta n = (\Delta n_1, \Delta n_2, \Delta n_3)$. Let these parameters be stacked into a single vector Δp given by:

$$\Delta p = \begin{pmatrix} \Delta s \\ \Delta \theta \\ \Delta n \end{pmatrix} \quad (28)$$

and let the matrix A be flattened to give a column vector

$$a = \begin{pmatrix} A_{11} & A_{12} & \dots & A_{33} \end{pmatrix}^T \quad (29)$$

The Jacobian of a w.r.t. to the parameter vector Δp can then be written as :

$$J_{a|p} = \begin{pmatrix} n_1 & 0 & 0 & | & 0 & 0 & 0 & | & s_1 & 0 & 0 \\ n_2 & 0 & 0 & | & 0 & 0 & -1 & | & 0 & s_1 & 0 \\ n_3 & 0 & 0 & | & 0 & 1 & 0 & | & 0 & 0 & s_1 \\ - & - & - & | & - & - & - & | & - & - & - \\ 0 & n_1 & 0 & | & 0 & 0 & 1 & | & s_2 & 0 & 0 \\ 0 & n_2 & 0 & | & 0 & 0 & 0 & | & 0 & s_2 & 0 \\ 0 & n_3 & 0 & | & -1 & 0 & 0 & | & 0 & 0 & s_3 \\ - & - & - & | & - & - & - & | & - & - & - \\ 0 & 0 & n_1 & | & 0 & -1 & 0 & | & s_3 & 0 & 0 \\ 0 & 0 & n_2 & | & 1 & 0 & 0 & | & 0 & s_3 & 0 \\ 0 & 0 & n_3 & | & 0 & 0 & 0 & | & 0 & 0 & s_3 \end{pmatrix} \quad (30)$$

Also, the Jacobian of the flow disparity $d = (u, v)^T$ w.r.t. vector a can be written as :

$$J_{d|a} = \frac{f}{D} \begin{pmatrix} -u' & uu'/f & -vu'/f & | & f & u & v & | & 0 & 0 & 0 \\ -v' & -uv'/f & -vv'/f & | & 0 & 0 & 0 & | & f & u & v \end{pmatrix} \quad (31)$$

where

$$D = A_{11}f + A_{12}u + A_{13}v \quad (32)$$

Then,

$$J_{d|p} = J_{d|a} J_{a|p} \quad (33)$$

Let the standard deviation in each of the parameters in Δp be given. The covariance matrix formed by these is denoted by Σ_p . Then, the covariance of d is given, in terms of Σ_p , by the equation :

$$\Sigma_d = J_{d|p} \Sigma_p J_{d|p}^T \quad (34)$$

The covariance Σ_d of the flow measures the uncertainty of the optical flow. It is a function of the image coordinates u and v . It was observed for the image in the sequence of the truck crossing the runway, that the covariance changed more significantly in the vertical v direction

in the images. This is expected in this type of images since the range of the objects sharply decreases in the lower parts of the images. If one neglects the correlation between the d_u and d_v components of the flow, the matrix Σ_d can be approximated as a diagonal matrix

$$\Sigma_d \simeq \begin{pmatrix} \sigma_u^2 & 0 \\ 0 & \sigma_v^2 \end{pmatrix} \quad (35)$$

The plot of σ_u and σ_v against the row value v (keeping column value constant $u = 256$) is shown in Fig. 10(a). It can be seen that the sensitivity increases as v increases. This is because the range x of the points on the runway plane decreases as the column coordinate increase.

A translating object on the ground is equivalent to a translation of the camera in the opposite direction. Hence, the residual disparity induced by the movement can be estimated by:

$$d = J_{d|p} \Delta p \quad (36)$$

Since only translation is considered, we can write:

$$d = J_{d|s} s_b \quad (37)$$

where $s_b = (s_{b1}, s_{b2}, s_{b3})^T$ is the displacement of the moving obstacle in one frame.

In the case of the image sequence of the crossing truck, the elements of $J_{d|s}$, which is the flow induced by movement in of $1ft/frame$ in each direction x , y and z is plotted against the column value v keeping row value $u = 256$. The plots of the induced flows in u and v direction are shown in Fig. 10(b) and (c), respectively. Similar to the previous case, the induced optical flow increases with the increase in column coordinate, i.e. decrease in the range of the point.

Consider for example, an obstacle crossing the runway. Then, s_{b1} , s_{b3} are nearly zero and the disparity in this case is given by:

$$d = \frac{\partial d}{\partial s_2} s_{b2} \quad (38)$$

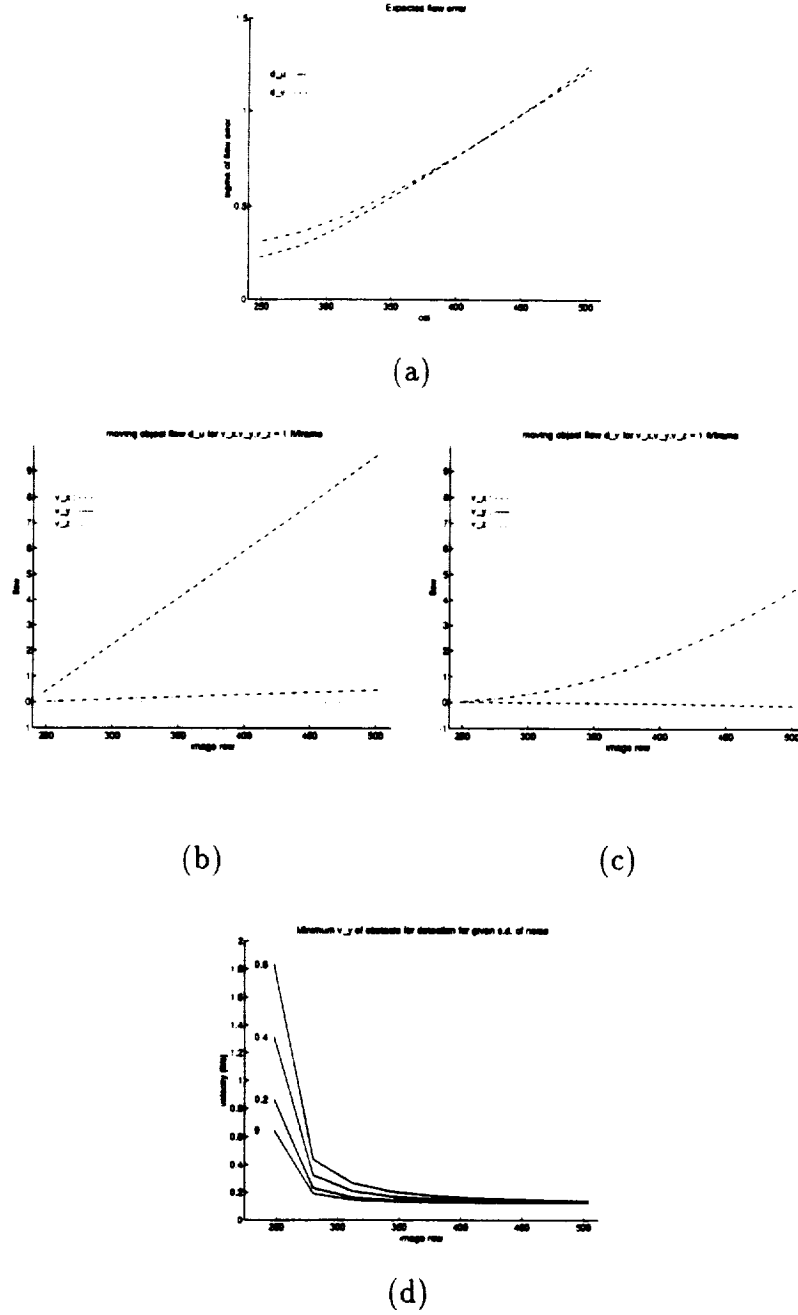


Figure 10: Sensitivity analysis of flow to camera and object motion parameters: (a) Plot of standard deviation of error in u and v directions against the v values for $u = 256$. (b),(c) Plots of the optical flow in u and v directions induced by the movement of obstacle in x, y, z directions u flow. (d) Plot of threshold velocity of an object against the v image coordinate for different values of the uncertainty in flow

In fact, the disparity will mostly be in the horizontal (u) direction. Hence, comparing σ_u with d_u , one can get the threshold velocity which can be detected at a given position in the image,

$$(s_b)_{thresh} = \frac{\sigma_u}{\partial d_u / \partial s_2} \quad (39)$$

If we also account for the uncertainty in the computation of the flow, the equation is approximately modified as:

$$(s_b)_{thresh} \simeq \frac{\sqrt{\sigma_u^2 + \sigma_{du}^2}}{\partial d_u / \partial s_2} \quad (40)$$

where σ_{du} is the uncertainty in the computation of the flow in u direction.

For the sequence of crossing truck, the value of $(s_b)_{thresh}$ is plotted against the column value in Fig. 10(d) for different values of σ_{du} . Fig. 11 shows the thresholding of the smoothed optical flows of Fig. 9. The obstacle, which is a moving truck, is successfully detected.

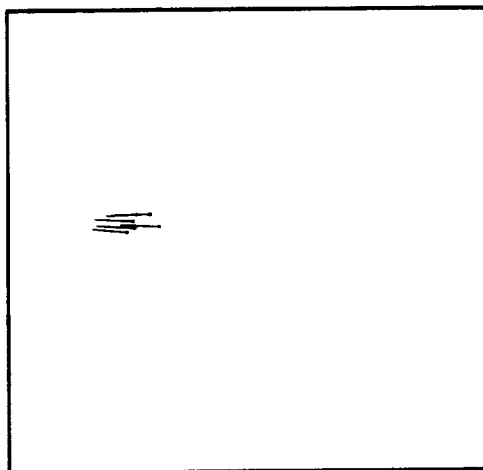


Figure 11: Thresholding of smoothed residual flows. The obstacle is detected.

8.2 Improvement of plane parameters

Since the obtained optical flows are the residual flows after warping using the given A parameters, they represent the error in the flow caused by the error in the A parameters. A Bayesian method can be used to improve the accuracy of A by applying an iterated least squared algorithm.

First note that in equation (12) the warping does not change if the matrix A is scaled by a constant. Thus, A actually has 8 instead of 9 independent parameters. Hence, in order to avoid singularities in the covariance matrices, one can scale a by setting its first element to 1 and scaling the rest of the elements appropriately. The resulting vector can be denoted by:

$$\hat{a} = \frac{1}{A_{11}} \begin{pmatrix} A_{12} & A_{13} & \dots & A_{33} \end{pmatrix}^T \quad (41)$$

The Jacobian of the flow w.r.t. this vector is given by $J_{d|\hat{a}}$. Since d is invariant to the scaling of a , the Jacobian can be found by extracting the columns of $J_{d|a}$ except for the first one, corresponding to A_{11} . The \hat{a} vector can then be improved by adding an increment of

$$\Delta \hat{a} = (J_{d|\hat{a}}^T \Sigma_d^{-1} J_{d|\hat{a}} + \Sigma_{\hat{a}0}^{-1})^{-1} J_{d|\hat{a}}^T \Sigma_d^{-1} d \quad (42)$$

where Σ_d is the covariance of the flow d and $\Sigma_{\hat{a}0}$ is the prior covariance of \hat{a} , given by:

$$\begin{aligned} \Sigma_{\hat{a}0} &= J_{\hat{a}|p} \Sigma_p J_{\hat{a}|p}^T \\ J_{\hat{a}|p} &= \left[\frac{1}{A_{11}} J_{a|p} - \frac{a}{A_{11}^2} J_{A_{11}|p} \right]_{col 2 \dots 8} \end{aligned} \quad (43)$$

i.e. columns 2 to 8 of the matrix in the brackets.

A new disparity can be found using the modified A and the procedure can be repeated until a satisfactory accuracy is obtained. Also, the inverse of the a-posteriori covariance of A is given by:

$$\Sigma_{\hat{a}}^{-1} = J_{d|\hat{a}}^T \Sigma_d J_{d|\hat{a}} + \Sigma_{\hat{a}0}^{-1} \quad (44)$$

Correction of the plane parameters was performed on the raw optical flows of the crossing truck sequence in Fig. 8(b) (using only five frames, without tracking). The residual disparities after correction are shown in Fig. 12(a). These disparities were thresholded using the same procedure as in the previous section. The result is shown in Fig. 12(b). The obstacle is successfully detected.

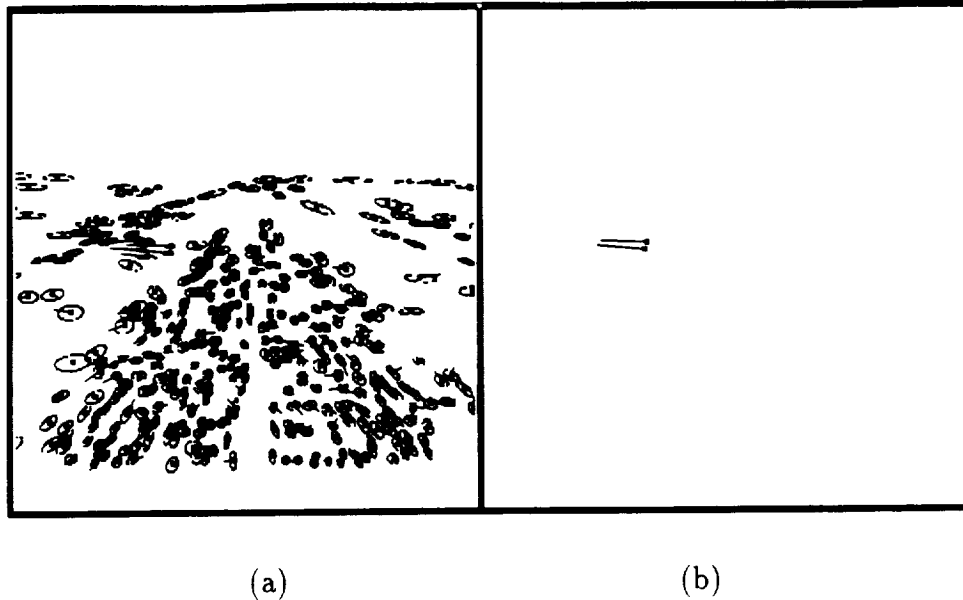


Figure 12: Correction of plane parameters: (a) Residual disparities after correction (b) Thresholding of smoothed residual flows. The obstacle is detected.

9 Separating Moving Objects

The above method detects two kinds of obstacles: moving obstacles as well as stationary obstacles at a height above the ground. In order to distinguish between the two, another constraint is required. Nelson [4] proposed a method using the constraint ray filtering for motion detection. It is based on the fact, noted in the context used by Thompson and Pong [15], that in a rigid environment, the projected 3-D velocity of any point in the image is constrained to lie on a 1-D locus in optical flow space whose parameters depend only on the

camera motion. Thus, in principle, if the motion field and camera motion are known, an independently moving object can be detected because its projected velocity is unlikely to fall on this locus.

$$\begin{pmatrix} d_u \\ d_v \end{pmatrix} = \frac{1}{x} T_s \Delta s + T_\theta \Delta \theta = \frac{1}{x} V_{yz} + V_r \quad (45)$$

where

$$\begin{aligned} T_s &= \begin{pmatrix} u & -f & 0 \\ v & 0 & -f \end{pmatrix} \\ T_\theta &= \begin{pmatrix} v & uv/f & -f(1 + u^2/v^2) \\ -u & f(1 + v^2/f^2) & -uv/f \end{pmatrix} \end{aligned} \quad (46)$$

and x is the range of the object.

The optical flow of a stationary object is therefore a linear function of $1/x$ and is constrained to lie on a single line in the $u - v$ plane as shown in Fig. 13. Furthermore, since the range of the object is bounded, the flow is actually constrained to a line segment. However, due to inaccuracies in estimation of the flow, as well as the linear and angular displacements, optical flow should lie within a *region* around the segment. The optical flow of the feature of a stationary object at *any* height has to satisfy the Nelson's constraint [4]. If the constraint is not satisfied, one can conclude that the object is moving. This constraint was successfully used by Tang and Kasturi [13, 14] to separate moving objects from background in our previous work.

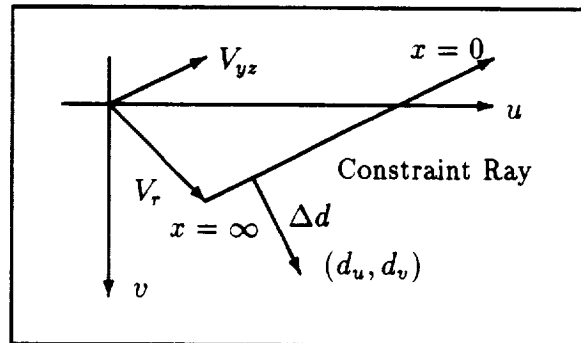


Figure 13: Constraint ray for x ranging from 0 to ∞

Note that, if residual disparities are used instead of the total disparities, the rotational motion is fully compensated, and the constraint line passes through the origin. The expression of the constraint becomes:

$$\begin{pmatrix} d_u \\ d_v \end{pmatrix} = \left(\frac{1}{x} - \frac{1}{x_p} \right) T_s \Delta s = \left(\frac{1}{x} - \frac{1}{x_p} \right) V_{yz} \quad (47)$$

where

$$x_p = \frac{f}{n_1 f + n_2 u + n_3 v} \quad (48)$$

is the range of a 3-D point on the runway plane which has the same image coordinates.

In order to apply the constraint, the residual flow is taken, and the shortest distance vector of the flow to the constraint segment, Δd is found in the $u - v$ plane. The covariance of the flow is taken as the sum of the covariances due to the error in estimation, and the covariance due to error in linear and angular displacements. Then, the weighted Mahalanobis distance $\Delta d \Sigma_d \Delta d$ is thresholded in order to locate potential moving obstacles.

The minimum velocity of the moving object can then be determined by solving the following equations

$$\Delta d = \frac{\partial d}{\partial s_b} \Delta s_b, \quad n^T s_b = 0 \quad (49)$$

where s_b is the velocity vector of the object moving on the ground, perpendicular to the normal vector. Note that the velocity could be greater than this value, since we have only found the minimum distance, but the actual object could be at a different range.

Also, it should be noted that only the obstacles which have a cross component of velocity can be detected. Moving obstacles moving in the same direction as the aircraft, i.e. parallel to the runway cannot be distinguished from a stationary obstacle at a height, due to the inherent ambiguity in monocular image sequences. Also, if a point is moving with such a velocity, that the residual disparity due to the height exactly compensates the disparity due to its independent motion, then the point would not be detected. In order to resolve these problems, stereo image sequences are required.

Nelson's constraint was applied on the smoothed velocity estimates of Fig. 9. The shortest distance measure Δd , which show the deviation from Nelson's constraint are shown in

Fig. 14(a), along with the covariance ellipses. Thresholding was done on these, showing the detected truck in Fig. 14(b).

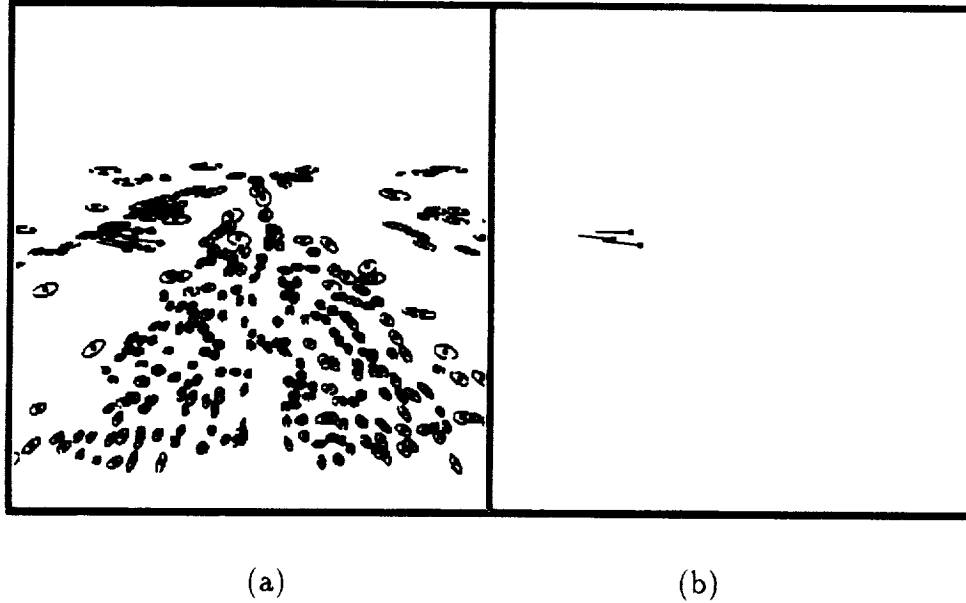
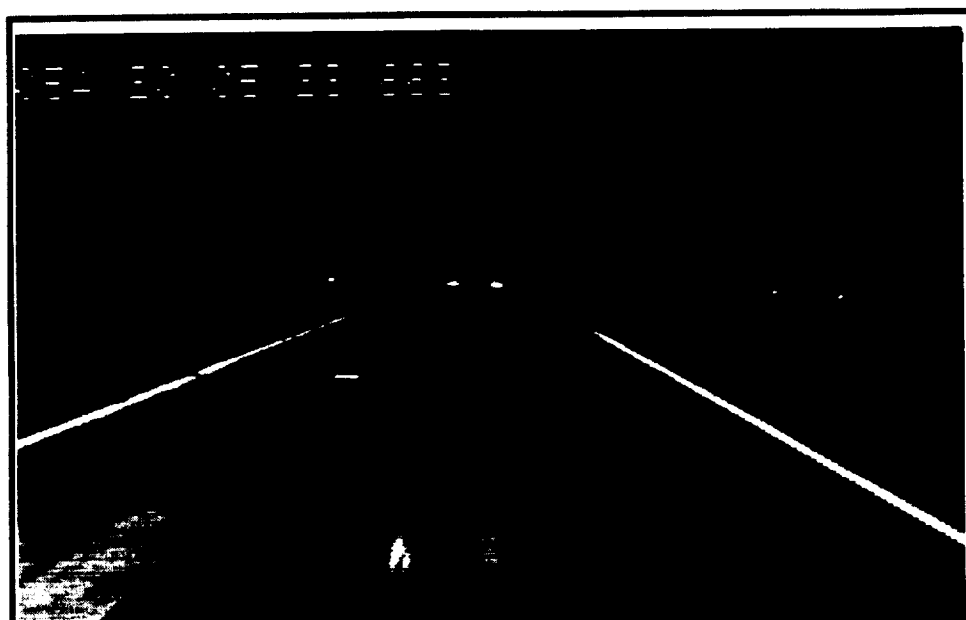


Figure 14: Application of Nelson's constraint : (a) Shortest distance measure Δd along with estimated covariance showing the deviation from Nelson's constraint. (b) Thresholding of the shortest distance. Moving truck is detected.

10 Observations and results

The method described above was also applied to other video image sequences supplied by NASA. In Fig. 15, frame number 70 of the image sequence *converging_truck_new* of a truck moving along the runway towards the helicopter is shown.

Optical flow estimation was first done using the minimum number (five) of frames. A zoomed part of the original image is shown in Fig. 16(a). The raw feature flows are shown in Fig. 16(b). The plane parameters were estimated using these flows and the flows were corrected using the improved parameters. The resulting flows are shown in Fig. 16(c). In both figures, the estimated covariance ellipses are also shown, magnified 5 times for clarity while the flow vectors are magnified 25 times. The feature flows were then thresholded using



the Mahalanobis norm w.r.t their covariances. The truck was easily detected as shown in Fig. 16(d).

To verify the tracking of features, fifteen frames were used, with frame 70 as the center frame. A zoomed part of the original image is shown in Fig. 17(a). The tracked features and their smoothed velocity estimates are shown in Fig. 17(b) and Fig. 17(c), respectively. The covariance ellipses are also shown and the magnifications are the same as in the previous case. The results after thresholding in a similar manner are shown in Fig. 17(d).

Nelson's constraint was applied on the smoothed velocity estimates of Fig. 17. The shortest distance measure Δd , which show the deviation from Nelson's constraint are shown in Fig. 18(a), along with the covariance ellipses. Thresholding was done on these. However, the converging truck was not detected as shown in Fig. 18(b). This could be because the truck is moving towards the helicopter, hence the disparity due to the motion may be compensated by the disparity due to the height of the truck.

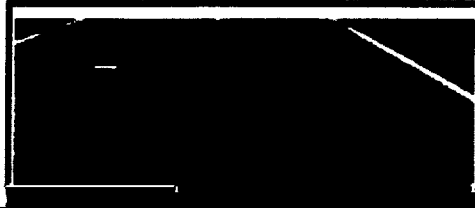
Similar experiments were performed on the image sequence *odd_trucks_new* containing a number of stationary trucks. Frame number 200 of this sequence is shown in Fig. 19. Results of using five frames, and fifteen frames are shown in Fig. 20 and Fig. 21, respectively in same manner as the previous image sequence. In this sequence, only the truck which was the nearest to the helicopter is detected, whereas the other trucks are not. Also, there are several false alarms. Application of Nelson's constraint on smoothed residual optical flows is shown in Fig. 22. It is seen that the results of this sequence are not as good. Even though the large truck is stationary, it is detected as a moving obstacle. A possible explanation for the unsatisfactory performance of our algorithm on this sequence is that the trucks in this sequence are quite large, and many features which are detected as corners are not really corners but parts of the edge. Due to the aperture effect, the value of the optical flow may be unreliable.

(a)

(b)



(a)



(b)



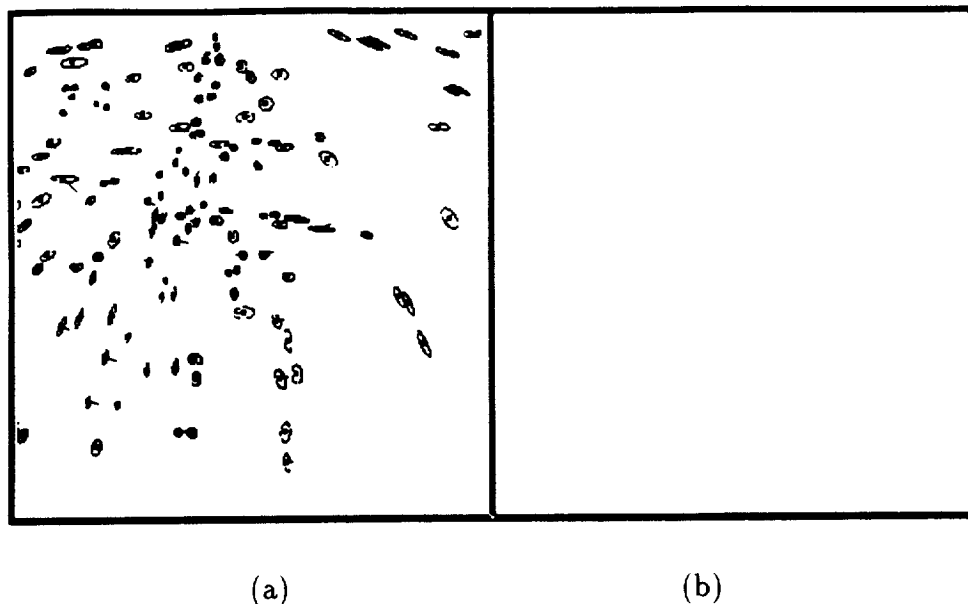
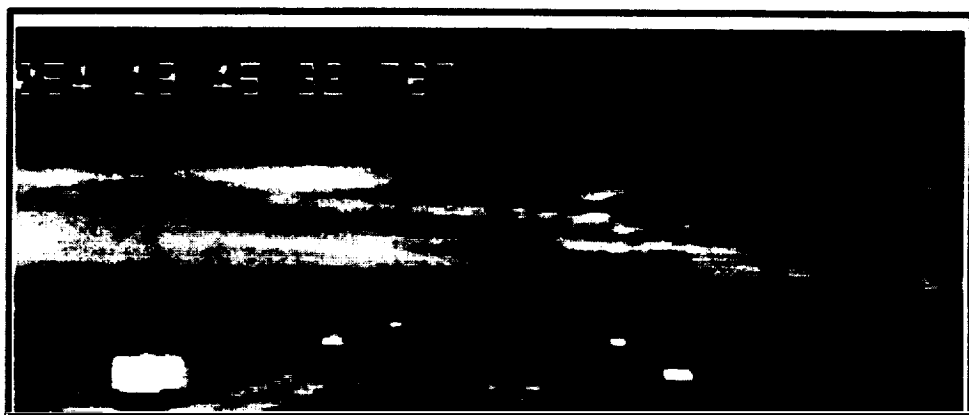


Figure 18: Application of Nelson's constraint : (a) Shortest distance measure Δd along with estimated covariance showing the deviation from Nelson's constraint. (b) Thresholding of the shortest distance. Truck is not detected, since it is moving *along* the runway.

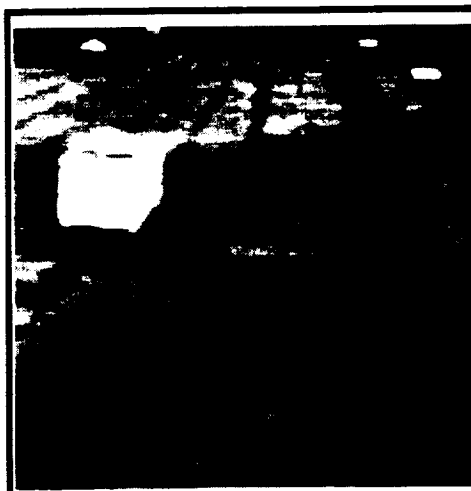
11 Summary and Future Work

In this report, we proposed a method for detection and tracking of obstacles on a runway for autonomous navigation and landing of aircrafts. Significant features were detected on the runway and residual optical flow was obtained. Warping was performed so that the ego-motion of the features was compensated as much as possible. However, due to the inaccuracy of the given parameters, full compensation was not possible. Hence, a sensitivity analysis of the warping to the variation of the given parameters was studied. A method to improve the accuracy of the warping using the residual disparities was also proposed. Application of this method to several runway image sequences was demonstrated and it was shown that obstacles can be separated from numerous extraneous features such as tire-marks.

Future work includes improvement and testing of each of the above stages for different image sequences under various conditions. An attempt is being made to use the normal optical flows, which are easier to find than the full optical flows, for correction of plane



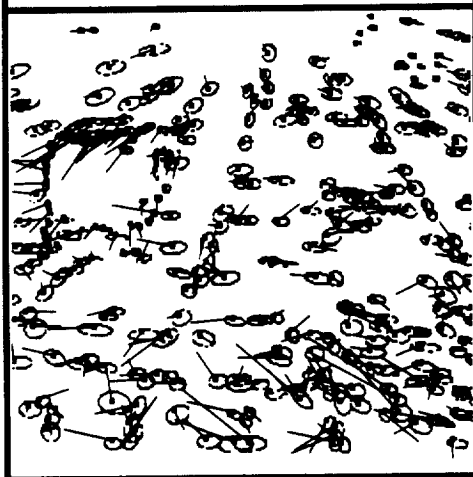
(a)



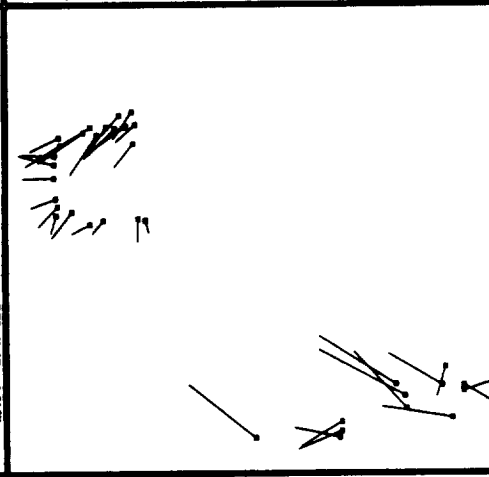
(b)



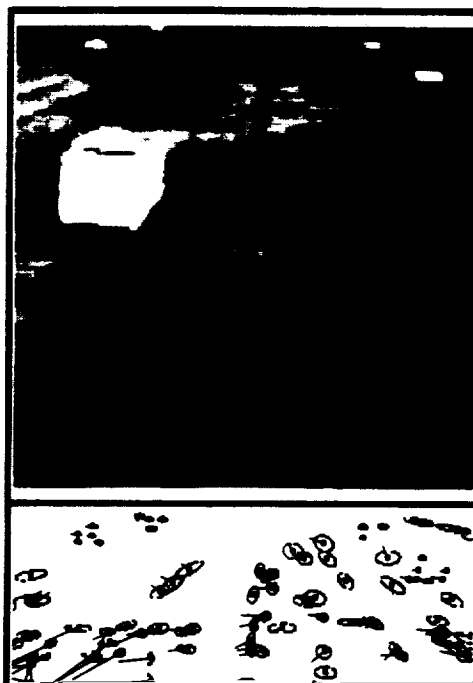
(c)



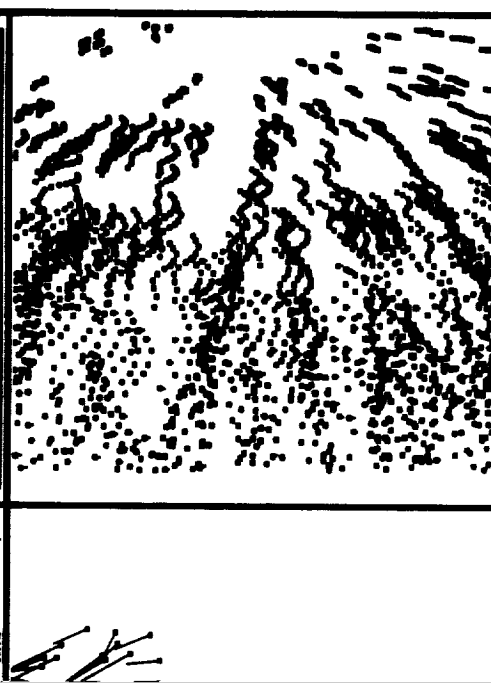
(d)



(a)



(b)



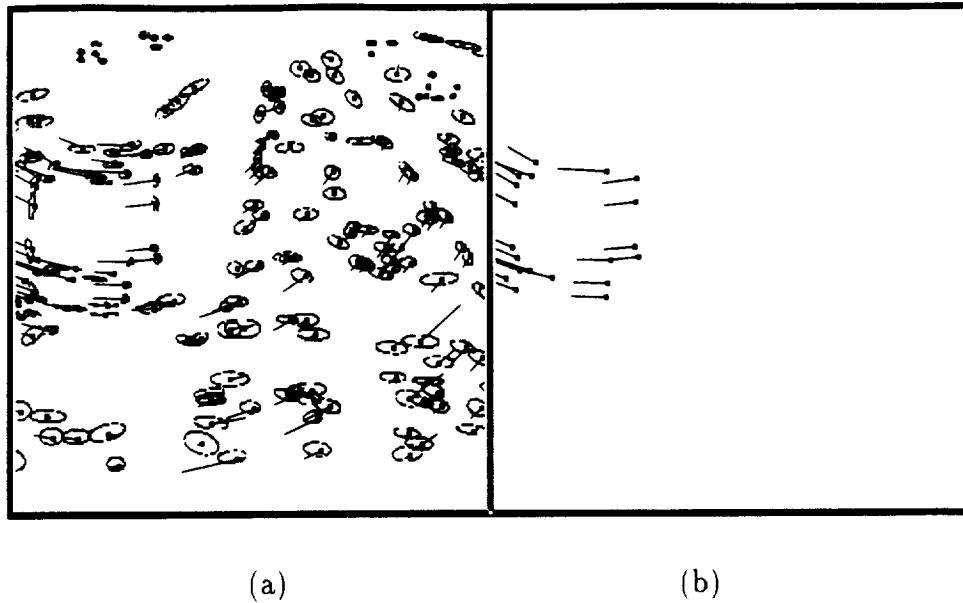


Figure 22: Application of Nelson's constraint : (a) Shortest distance measure Δd along with estimated covariance showing the deviation from Nelson's constraint. (b) Thresholding of the shortest distance.

parameters. Also, sensor fusion methods to combine the information available from different sources such as the GPS, the INS, and landmarks like lines and beacons on the runway, are being explored.

In a different approach that we are currently pursuing, the runway is assumed to be piecewise planar and regions violating the planar constraint are assumed to be obstacles. A recursive motion based segmentation algorithm is being developed to segment the image into different regions. Both linear and nonlinear estimation methods are being evaluated. Knowledge of the camera motion will be used to distinguish image regions due to moving objects from those due to static background. In this work objects are assumed to be rigid and moving on the runway. A constraint based solution approach using the knowledge of the camera motion and the estimated runway plane parameters will be used to incrementally estimate the object motion in the world coordinate system. Since it will not be possible to compute an initial solution due to the nonzero object height, multiple solutions will be hypothesized assuming that some of the object features are at zero height. Hypothesized

solutions will then be tracked in every frame using a Kalman filter and will be verified by enforcing the rigidity constraint. Solutions violating this constraint will be dropped. After several frames have elapsed, we expect that all the false solutions will disappear and only a unique solution converging to the true value will remain.

References

- [1] J.L. Barron, D.J. Fleet, S.S. Beauchemin, and T.A. Burkitt. Disparity analysis of images. In *Proc. IEEE Computer Vision and Pattern Recognition*, volume 2, pages 236–242, 1992.
- [2] B.K.P. Horn and B.G. Schunck. Determining optical flow. *Artificial Intelligence*, 17:185–203, 1981.
- [3] B. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *DARPA Image Understanding Workshop*, pages 121–130, 1981.
- [4] Randall C. Nelson. Qualitative detection of motion by a moving observer. In *Proc. IEEE Computer Vision and Pattern Recognition*, pages 173–178, 1991.
- [5] Eero P. Simoncelli, Edward H. Adelson, and David J. Heeger. Probability distributions of optical flow. In *Proc. IEEE Computer Vision and Pattern Recognition*, pages 310–315, 1991.
- [6] Philip N. Smith. Nasa image data base user's guide. Technical report, NASA Ames Research Center, Moffett Field, CA, 1990.
- [7] S.M. Smith. A new class of corner finder. In *Proc. 3rd British Machine Vision Conference*, pages 139–148, 1992.
- [8] S.M. Smith and J.M. Brady. SUSAN - a new approach to low level image processing. *Int. Journal of Computer Vision*, 1996. In publication.

- [9] Banavar Sridhar and Anil V. Phatak. Analysis of image-based navigation system for rotocraft low-altitude flight. *IEEE Trans. on Systems, Man and Cybernetics*, 22, March/April 1992.
- [10] Banavar Sridhar, R. Suorsa, and B. Hussien. Passive range estimation for rotocraft low-altitude flight. *Machine Vision and Applications*, 6:10-24, 1993.

