Tracking of moving objects in scenery using subspace projection using independent component analysis

Brian J. Noe
Fredric M. Ham
Abstract

A system is developed for tracking moving objects through natural scenery. A technique is presented for performing change detection on imagery to determine the difference between two images or a sequence of images. From there an algorithm is presented to detect the presence of a new object and/or the deletion of objects. Then the application of a Variable Structure Interacting Multiple Model tracking filter is presented.

The method of performing change detection is based upon the concept of image subspace projection. A set of “basis” image maps are formed when combined with a mixing matrix can recreate the original image. The subsequent images are then projected into the base image. The projected image is then subtracted from the original image to perform the change detection. Spatial Filtering is applied to increase the contrast between the change and the background then an adaptive filter is then applied to pass the locations of changes in the images into the tracking filter. Tracking is performed through the use of multiple motion models. The filter’s motion models are adaptive added or deleted as required by the moving object’s dynamics. The moving object’s state is estimated through extended Kalman filtering.

Keywords: Change Detection, Independent Component Analysis, Tracking, Interacting Multiple Models

Introduction

Tracking of moving objects through a series of images is motivated by several diverse requirements. Modern image compression techniques, such as MPEG-2, use the information of changes between frames to delete information for compression. Military and civilian surveillance looks for changes in scenery to determine construction activity, accrual of assets, crop health and troop movement.

The majority of change detection algorithms for imagery require a differencing of recorded intensity or phase values between images. For these algorithms to properly function, geometric alignment must be performed between the images. For moving targets this is not always possible because the translation of the target may be greater than the possible image size or for surveillance aircraft they may not be able to image the scene from exactly the same location at different time. When the images are not properly aligned, cropping is usually performed to manually align the images to perform change detection. The technique presented does not require that the images be perfectly aligned. The algorithm has proved to be invariant to both translation and rotation.
Overview of Independent Component Analysis

A fundamental problem is signal analysis, is to decompose noisy measurements into a mixture of primitive signals of which the primitive signals and a mixing matrix are known or to be estimated. Another way to express the problem is to identify the basic signals from which a complex signal is composed. This is much akin to Fourier analysis, however the primitive signals of interest are not assumed to be sinusoids. In real life the signals encountered tend to be non-gaussian and non-linear. Independent Component Analysis is a statistical regression technique that estimates a mixing matrix and a set of independent “basis” vectors for a series of measurements.

In Independent Component Analysis we assume to have a series of $n$ measurements from $m$ sensors that we use to form measurement matrix $x$. We then wish to estimate a set of basis vectors $s$, that are as independent as possible and a mixing matrix $A$ that is used to reconstruct the measurements.

$$x = As$$ (1)

The starting point for solving this transformation is that the independent components are statistically independent and that the independent components have non-gaussian distributions. We can assume for simplicity that the mixing matrix $A$ is square. The problem of finding the independent components can be examined as finding a separating matrix $W$, $W = \text{pseudo-inverse} (A)$, that we can use to separate the independent components from the data.

$$s = Wx$$ (2)

Before the FastICA algorithm can be applied to solve for the separation matrix, the data should be preprocessed by centering and whitening the data. To center the data, the mean is subtracted from the data. Whitening the data is accomplished by linearly transforming the data so the its components are uncorrelated and of unit variance. Another way to express this is that the covariance of the data equals the identity matrix.

$$E\{xx^T\} = I$$ (3)

Eigen-value decomposition of the covariance matrix $E\{xx^T\} = E\text{DE}^T$, where $E$ is the orthogonal matrix of eigenvectors and $D$ is a diagonal matrix of eigenvalues may be used to whiten the data as:

$$\tilde{x} = E D^{-1/2} E^T x$$ (4)

Whitening the data also has the effect of transforming the mixing matrix $A$ as

$$\tilde{x} = E D^{-1/2} E^T As = \tilde{A}s$$ (5)

It can be shown that the new mixing matrix is orthogonal.
and that whitening reduces the number of parameters to be estimated from \( n^2 \) to \( n(n-1)/2 \).

Many techniques have been described to find the separation matrix \( w \). The method chosen is the FastICA algorithm by Aapo Hyvarinen and Erkki Oja that is presented. The FastICA algorithm is fixed-point algorithm for finding the maximum nongaussianity of \( w'x \) as measured by the negentropy \( J(w'x) \). The FastICA algorithm may also be derived as a Newton iteration.

The derivation of FastICA is given in and is repeated here for convenience.

First note that the maxima of the appropriate negentropy of \( w'x \) are obtained at certain optima of \( E(G(w'x)) \). According to Kuhn-Tucker conditions, the optima of \( E(G(w'x)) \) under the constraint \( E((w'x)^2) = ||w||^2 = 1 \) are obtained at points where

\[
E{\{xg(w^T x)\}} - \beta w = 0
\]  

(7)

Let us try to solve the equation by Newton's method. Denoting the function on the left-hand side by \( F \), we obtain its Jacobian matrix \( JF(e) \) as

\[
JF(w) = E{xx^Tg'(w^T x)} - \beta I
\]  

(8)

To simplify the inversion of the matrix, we decide to approximate the first term in the above. Since the data is sphered, a reasonable approximation seems to be \( E{xx^Tg'(w^T x)} = E{xx^T} E{g'(w^T x)} = E{g'(w^T x)}I \). Thus the Jacobian matrix becomes diagonal, and can easily be inverted. Thus we obtain the following approximate Newton iteration:

\[
w^+ = w - [E\{xg(w^T x)\} - \beta w]/[E(g'(w^T x)) - \beta]
\]  

(9)

\[
w^* = w^+ / \|w^+\|
\]  

(10)

where \( w^* \) denotes the new value of \( w \), \( \beta = E\{w^T x g'(w^T x)\} \), and the normalization has been added to improve stability. This algorithm can be further simplified by multiplying both sides of the equation by multiplying both sides by \( \beta \)- \( E(g'(w^T x)) \). This gives the following fixed-point algorithm

\[
w^+ = w - [E\{xg(w^T x)\} - [E(g'(w^T x))w]
\]  

(11)

\[
w^* = w^+ / \|w^+\|
\]  

(12)

The above derivation also enables a useful modification of the fixed-point algorithm. It is well known that the convergence of the Newton method may be rather uncertain. To ameliorate this, one may add a step size obtaining a stabilized fixed-point algorithm.
\[ w^* = w - \mu [E\{x^T g(w^T x)\} - \beta w] [E\{g'(w^T x)\} - \beta] \]  

\[ w^* = w^* / \|w^*\| \]  

(13)  
(14)

Were \( \beta = E\{ w^T x g(w^T x)\} \) as above and \( \mu \) is a step size parameter and may change with the iteration count. Taking a \( \mu \) that is much smaller the unity (say 0.1 or 0.01), the algorithm converges with much more certainty.

**Differencing through Subspace Projection**

If we consider a system that takes “snap shots” of scenery over time, we wish to develop a system model that can detect the changes between any two images. The images acquired by a digital camera (Infrared or Optical) or through a radar (SAR images) can be considered 2-dimensional images where each pixel represents an discrete, quantized, intensity value (in the case of color there are three such values at each pixel). When successive images are taken of the same area, the second image can be thought of a linear combination of the convolution of the first image and the changed caused by foreign objects or movement in the scene. Mathematically this can be represented as:

\[ f_2(x, y) = f_1(x, y) * h(x, y) + f_e(x, y) \]  

(15)

Where \( * \) donates convolution. The function \( h(x, y) \) is the 2-dimensional impulse response, which represents the relative shift and blurring in the image due to slight motion or change in the Point Spread Function of the sensors. This model implies that the second image is linearly related to the first image or a shifted version of it.

In equation 1 both \( h(x, y) \) and \( f_e(x, y) \) are unknown. To solve for both \( h(x, y) \) and \( f_e(x, y) \), \( h(x, y) \) is modeled as a finite two-dimensional discrete filter of size \((N_x, N_y)\). If we choose both \( N_x \) and \( N_y \) as odd integers and \((n_x, n_y) = (N_x/2 - 0.5, N_y/2 - 0.5)\), then equation 1 can be rewritten as

\[ f_2(x, y) = \sum_{m=-n_x}^{n_x} \sum_{n=-n_y}^{n_y} h_{mn} f_1(x - m\Delta_x, y - n\Delta_y) + f_e(x, y) \]  

(16)

Where \((\Delta x, \Delta y)\) is the sensor sample spacing in the \((x, y)\) domain. If we minimize the following error function to solve for \( h_{mn} \)

\[ \sum_i \sum_j | f_2(x_i, y_j) - \sum_{m=-n_x}^{n_x} \sum_{n=-n_y}^{n_y} h_{mn} f_1(x_i - m\Delta_x, y_j - n\Delta_y) |^2 \]  

(17)

The resultant solution is used to estimate \( f_2(x, y) \) via

\[ \hat{f_2}(x, y) = \sum_{m=-n_x}^{n_x} \sum_{n=-n_y}^{n_y} \hat{h}_{mn} f_1(x - m\Delta_x, y - n\Delta_y) + f_e(x, y) \]  

(18)

From this the statistic used to detect a change between the two images is...
The signal $\hat{f}_2(x_i, y_i)$ is the projection of $f_2(x_i, y_i)$ into a linear subspace defined by $f_1(x_1, y_1)$ and $N-1$, where $N = N_x N_y$ or its shifted versions. This now reduces the problem to finding a signal subspace for $f_1$ and forming the projection of $f_2$ into the subspace defined by $f_1$. Then subtracting this projection from the original $f_2$ to find the changes between $f_2$ and $f_1$.

Derivation of Change Detection through Image Subspace Projection

Change detection through image subspace projection is a differencing technique that detects changes in the projection of one image into the subspace of another image from the original image. The technique employs calculating a subspace basis for an image and projecting a second image into the subspace of the first image. This projection is then subtracted from the original second image to determine the differences between the two images.

Step 1. Sample Image 1

A measurement matrix $X_1$ is created from the first image by sampling $n \times n$ non-overlapping sub-arrays of the first image. The size of the sub-array, $n$, is user selectable but must be an integral divisor of the overall size of the image. Once the sub-arrays are formed, the columns of the sub-array are "stacked" on top of one another to form a column vector of length $n \times n \times 1$. These column vectors are used to form the columns of $X_1$.

Step 2. Perform fastICA algorithm on $X_1$ to generate $A_1$ and $S_1$.

On the measurement matrix $X_1$, the fastICA algorithm is performed to determine the estimates of the Independent Components $S_1$ and the mixing matrix $A_1$. The independent Components $S_1$ form the subspace vectors (sub-image arrays) for the first image.

Step 3. Sample Image 2

The measurement matrix $X_2$ is created from the second image. The sub-array size, $N$, is the same size as used on image 1.

Step 4. Project $X_2$ onto $S_1$ to find $A_{21}$.

The measurement matrix $X_2$ is projected onto the image subspace $S_1$ to find the mixing matrix of $X_2$ onto $S_1$ as

$$A_{21} = X_2 \ast \text{pinv}(S_1)$$

(20)

Where pinv is the pseudo inverse operation.
Figure 1 Example of Image processing to form measurement matrix

512 pixels

8 pixels

512/8 times 521/8 (4096) sub-arrays

Extracted n by n sub-array

8 pixels

4096 columns

512 pixels

Column Vector

64 by 1

\( \mathbf{X} = \begin{pmatrix} \cdots \end{pmatrix} \)

64 rows

4096 columns
Step 5. Calculate the projection of image 2's measurement data into image 1's subspace.

An estimate of image 2, $X_{2\hat{}}$, is formed from the mixing matrix $A_{21}$ and the original image subspace $S_1$.

$$X_{2\hat{}} = A_{21} \cdot S_1 \quad \text{(21)}$$

Step 6. Subtract the projection of image 2 in image 1's subspace form the original image 2.

The difference between the projection of second image and the second image is calculated.

$$\text{Difference} \_ \_{X1} \_ \text{to} \_ \_{X2} = X_2 - X_{2\hat{}} \quad \text{(22)}$$

This difference is then reformatted into n x n sub-arrays, these sub-arrays are then used to form a difference image.

Step 7. Spatial filter sharpen images.

A spatial filter is applied to sharpen the difference between the image and the background noise. The form of the spatial filter is visually scene as

$$f(x,y) = \begin{bmatrix} -1 & 4 & -1 \\ -1 & 0 & -1 \end{bmatrix}$$

or mathematically as $f(x,y) = 4 \cdot \text{Image}(x,y) - f(x-1,y) - f(x+1,y) - f(x,y-1) - f(x,y+1)$

$$\quad \text{(23)}$$


Because the images can be made from two un-correlated sensors or the images were taken at different times, the background "noise" is typically different between the images. To filter the data to find when "strong" changes have occurred, a +/- 3 sigma band rejection filter is adaptively calculated to find where the greatest changes have occurred. The mean and standard deviation of the difference image is calculated. A test is made to find all values that are 3 times the standard deviation above the mean and below the mean.

Step 9. Target Decalration

To track moving targets through imagery, a point grouping function is then performed to group connecting points that pass the adaptive threshold into one target declaration. The pixel location (i.e. the x and y position) of the target is determined from the point groups. This information along with the estimated vehicle heading direction (derived from the length to width profile) is fed into a tracking algorithm along with whether this is a high detection (new location) or low detection (left location) for a target.

Test Results

A moving target was recorded with a CCD video camera as it traversed down a road. The camera's position was both linearly translated and rotated over the recording interval to determine the amount of variance the algorithm could withstand and still perform change detection. An example of the ability of the algorithm to withstand both linear translation and rotations is presented below. The first two figures, Figure 2 and Figure 3 show two images a few seconds...
apart from the video. Figure 4 show the differences between he images when the second image is subtracted from the first. Figure 5 then shows the difference between the two images when subspace projection is employed to perform the change detection.

Figure 2 Image 1
Figure 3 Image 2

Figure 4 Simple Difference between Image 1 and Image 2

Notice shift in light post and mailbox.

Dark spot from Change in shadow

Proc. SPIE Vol. 4380 505
To investigate the ability to perform long term tracking of objects through scenery, a vehicle target moving down a road was recorded for a period of over one minute. The change detection algorithm was performed on a sub sample of the images recorded. Change detection was performed between every 15 frames of data, or about one a second. The results of the individual change detections were feed into the spatial filtering, adaptive thresholding, point grouping and centroid calculations to produce a target declaration. To perform tracking over an extended period, a coordinate conversion between the pixel mapping of the points and a local coordinate frame of reference had to be made. To convert from pixel dimensions to physical dimensions the distance to various objects in the scene were made manually.

The target indicator reports from section were feed into the tracker describer in \textsuperscript{1}. The calculated centroid positions were used as the position reports and the covariance was set as $R_{xx} = R_{yy} = 10$ pixels and $R_{xy} = R_{yx} = 0$ pixels (no cross coupling between the measurement errors). The tracker results from the given scenario were that a moving ground track was formed on the vehicle and the track persisted throughout the entire duration of the scenario. A second track was formed later in the scenario as a large object passed within the field of view of the camera. Figures 6 depicts the results of the number of tracks for the scenario.
Conclusions

Analysis of results show that the algorithm can withstand changes in background information of approximately 15% from either linear or rotational translation and still perform change detection and only detect moving targets and not the difference in the background.

The performance of change detection through subspace project is a powerful technique that eliminates the need to properly align and crop images. Independent Component Analysis provides an efficient and flexible means of finding the basis images to perform the subspace projection.

References


