FEATURE BASED IMAGE MATCHING FOR AIRBORNE PLATFORM

[1]HUMERA SIDDIQUA, [2]A.H.SHANTHAKUMARA, [3]MD. SHAHID

[1]M. Tech(CNE), [2]Asst Professor , (Computer Science), SIT , Tumkur, Scientist 'E', ADE, DRDO
E-mail: humera.siddiqua928@gmail.com, shanthakumara@sit.ac.in, mshahid79@gmail.com

Abstract- In computer vision, detection and tracking of targets is very complex problem and demands sophisticated solutions. Unmanned Aerial Vehicles (UAVs) are increasingly being used for reconnaissance and Surveillance. This framework mainly consists of image matching for reconnaissance and Surveillance. This framework mainly consists of image matching for Digital Scene Matching Area Correlation (DSMAC), Image Registration by using various correlation techniques, Mean Shift Algorithm, Cam shift Algorithm, Otsu Thresholding and Principal Component Analysis (PCA) and using Point detectors like Scale Invariant Feature Transform (SIFT) , Speed Up Robust Feature (SURF) and Saliency Map".

Keywords- Correlation, Cam shift algorithm, DSMAC, Image registration, Mean shift algorithm, Otsu thresholding, PCA, Saliency Map, SIFT, SURF.

I. INTRODUCTION

Visual detection and tracking is one of the most challenging issues in computer vision. Application of the visual detection and tracking are numerous and they span a wide range of applications including surveillance system, vehicle tracking and aerospace application, to name a few. Detection and tracking of abstract targets (e.g. vehicles in general) is a very complex problem and demands sophisticated solutions using conventional pattern recognition and motion estimation methods. Motion-based segmentation is one of the powerful tools for detection and tracking of moving targets. It is simple to detect moving objects in image sequences obtained by stationary camera.

DSMAC is a terminal stage guidance system where terminal phase area imagery is compared with preloaded satellite imagery that the low flying high-speed aerial vehicle carries in its memory. DSMAC is computationally intensive process requires high speed correlation between reference and aerial images. The accuracy of position estimation is entirely dependent on the image registration technique used to locate the aerial image on vast area of reference image.

Image registration is the process of transforming different sets of data into one coordinate system. Data may be multiple photographs, data from different sensors, times, depths, or viewpoints. It is used in computer vision, medical imaging, military automatic target recognition, and compiling and analyzing images and data from satellites.

Registration is necessary in order to be able to compare orintegrate the data obtained from these different measurements. In this we have implemented algorithms like Intensity-based method which register entire images or sub images. If sub images are registered, centers of corresponding sub images are treated as corresponding feature points, Feature-based methods which establish a correspondence between a number of especially distinct points in images, Spatial method operates in the image domain, matching intensity patterns or features in images and Frequency-domain method find the transformation parameters for registration of the images while working in the transform domain.

1.1 Correlation Techniques

The correlation operation is the heart of a correlation tracking system. The performance of such a system is dependent on the correlator used.

The correlator should produce a correlation surface with a strong response at the location of the target object while suppressing the response of false targets. The correlation between two signals (cross correlation) is a standard approach to feature detection as well as a component of more sophisticated techniques. This correlation process can be performed in the spatial domain or in the frequency domain. Normalized cross correlation has been computed in the spatial domain for this reason.

Spatial correlation can be performed as follows. Let the reference image be described by the vector R and the section of the current scene image by the vector S_{ij}. Then the correlation at each location (i, j) in the scene image is simply the dot product of two vectors, and is given by $C(i,j) = S_{ij} \times R.$

1.1.1. Template Matching by Cross-Correlation [1]

Cross-correlation for template matching is motivated by the distance measure (squared Euclidean distance)

$$d^2_{f,t}(u,v) = \sum_{x,y}[f(x,y) - t(x-u, y-v)]^2 \qquad (1)$$

Where, f is the image and the sum is over x, y under the window containing the feature t positioned at u, v.

$$d^2_{f,t}(u,v) = \sum_{x,y}[f^2(x,y) - 2f(x,y)t(x-u,y-v) + t^2(x-u,y-v)] \quad (2)$$

In the expansion of $d^2$the term $\sum t^2(x-u,y-v)$ is constant. If the term$\sum f^2(x,y)$ is approximately constant then the remaining cross-correlation term $c(x,y)$ is a measure of the similarity between the image and the feature.

$$c(u,v) = \sum_{x,y} f(x,y)t(x-u,y-v) \quad (3)$$

There are several disadvantages to using this for template matching for example, the correlation between the feature and an exactly matching region in the image may be less than the correlation between the feature and a bright spot.The correlation coefficient overcomes these difficulties by normalizing the image and feature vectors to unit length, yielding a cosine-like correlation coefficient

$$f(u,v) = \frac{\sum_{x,y}[f(x,y) - \bar{f}_{u,v}][t(x-u,y-v) - \bar{t}]}{\{\sum_{x,y}[f(x,y) - \bar{f}_{u,v}]^2 \sum_{x,y}[t(x-u,y-v) - \bar{t}]^2\}^{0.5}} \quad (4)$$

Where,$\bar{t}$is the mean of the feature and $f(u,v)$is the mean of $f(x,y)$ in the region under the feature. Equation (3) is referred as normalized cross-correlation.

1.1.2.Correlation Coefficient

2D correlation coefficient given by,

$$r = \frac{\sum_{x\in N}\sum_{y\in N}(A_{xy} - A')(B_{xy} - B')}{[\sum_{x\in N}\sum_{y\in N}(A_{xy} - A')^2(B_{xy} - B')^2]^{0.5}} \quad (5)$$

Where, A is a reference image or query image which is small than Search image B of given input image. A′ is a mean of A , B′ is a mean of B and r is the correlation coefficient range in the value -1.0 to +1.0.

Other correlation coefficient method [3] is given as,

$$S1(f,g) = \frac{\sum_{x\in N}\sum_{y\in N} f'(x,y)g'(x,y)}{[\sum_{x\in N}\sum_{y\in N}(f'(x,y)^2 * g'(x,y)^2)]} \quad (6)$$

Where, f(x, y) is a reference image or query image which is small than Search image g(x, y) in a given input image and N is the size of 'x' pixel value which is taken in range for f and g multiplied with size of 'y' pixel value which is taken in range for f(x,y) and g(x, y).In this,

$$f'(x,y) = [f(x,y) - \bar{f}],\ g'(x,y) = [g(x,y) - \bar{g}]$$
$$\bar{f} = \text{mean2}(f),\ \bar{g} = \text{mean2}(g)$$

1.1.3.Brunelli Messelodi -1 and Brunelli Messelodi -2 method [3]

In 1995 Brunelli Messelodi proposed a method-1 and method-2 for similarity and explanation for variables follows as above explained and it is given as,

$$S2(f,g) = 1 - \frac{\sum_{x\in N}\sum_{y\in N}|f'(x,y) - g'(x,y)|}{\sum_{x\in N}\sum_{y\in N}|f'(x,y)| + |g'(x,y)|} \quad (7)$$

$$S3(f,g) = \left(\frac{1}{N*N}\right)\frac{\sum_{x\in N}\sum_{y\in N}|f'(x,y) - g'(x,y)|}{\sum_{x\in N}\sum_{y\in N}|f'(x,y)| + |g'(x,y)|} \quad (8)$$

1.1.4.Khosravi and Schafer

In the context of mathematical morphology, Khosravi and Schafer[3] proposed the measure of similarity in 1996 as follows

$$S4(f,g) = \min_{x,y\in N}\{g(x,y) - f(x,y)\} - \max_{x,y\in N}\{g(x,y) - f(x,y)\} \quad (9)$$

1.2Kernel – based Tracking

1.2.1 Mean Shift Tracking Algorithm[4]

"Mean Shift is a tool for finding modes in a set of data samples, manifesting an underlying probability density function (PDF)."

Target representation

In object tracking, a target is usually defined as a rectangle or an ellipsoidal region in the image. Currently, a widely used target representation is the color histogram because of its independence of scaling and rotation and its robustness to partial occlusions. $\{x_i^*\}_{i=1\ldots n}$ is the normalised pixels in the target region, which is supposed to be centred at the origin point and have n pixels. The probability of the feature u (u = 1, 2, .. ., m) in the target model is computed as.

$$\begin{cases} \hat{q} = \{\hat{q}_u\}_{u=1\ldots m} \\ \hat{q}_u = C\sum_{i=1}^{n} k(||x_i^*||^2)\,\partial[bx_i^*) - u] \end{cases} \quad (10)$$

where $\hat{q}$ is the target model, $\hat{q}_u$is the probability of the uth element of $\hat{q}$, d is the Kronecker delta function, $b(x_i^*)$ associates the pixel x_i^* to the histogram bin and k(x) is an isotropic kernel profile. Constant C is a normalization function defined by

$$C = 1/\sum_{i=1}^{n} k(||x_i^*||^2) \quad (11)$$

Similarly, the probability of the feature u in the target candidate model from the candidate region centred at position y is given by

$$\begin{cases} \hat{p}(y) = \{\hat{p}_u\}_{u=1\ldots m} \\ \hat{p}_u(y) = C\sum_{i=1}^{n} k\left(\left\|\frac{y - x_i}{h}\right\|^2\right)\partial[b(x_i) - u] \end{cases} \quad (12)$$

$$C = 1/\sum_{i=1}^{n} k\left(\left\|\frac{y - x_i}{h}\right\|^2\right) \quad (13)$$

Where, $\hat{p}(y)$ is the target candidate model, $\hat{p}_u$(y) is the probability of the uth element of $\hat{p}(y)$, $\{x_i\}_{i=1\ldots n}$ are pixels in the target candidate region centred at y, h is the bandwidth and C is the normalization function which is independent of y.

In order to calculate the likelihood of the target model and the candidate model, a metric based on the Bhattacharyya coefficient is defined by using the two normalized histograms $\hat{p}(y)$ and $\hat{q}$ as follows

$$\rho[\hat{p}(y),\hat{q}] = \sum_{u=1}^{m}\sqrt{\hat{p}_u(y),\hat{q}_u} \quad (14)$$

Mean shift

Taylor expansion around $\hat{p}_u(y_0)$, the linear approximation of the Bhattacharyya coefficient $\rho[\hat{p}(y),\hat{q}]$ can be obtained as

$$\rho[\hat{p}(y),\hat{q}] \cong \frac{1}{2}\sum_{u=1}^{m}\sqrt{\hat{p}_u(y_0),\hat{q}_u} + \frac{C}{2}\sum_{u=1}^{n} w_i\, k\left(\left\|\frac{y-x_i}{h}\right\|^2\right) \quad (15)$$

Where,

$$w_i = \sum_{u=1}^{m}\sqrt{\frac{\hat{q}_u}{\hat{p}_u(y_0)}}\,\partial[b(x_i)-u] \quad (16)$$

The estimated target moves from y to a new position y1, which is defined as

$$y_1 = \frac{\sum_{i=1}^{n} x_i w_i g(\|(y-x_i)/h\|^2)}{\sum_{i=1}^{n} w_i g(\|(y-x_i)/h\|^2)} \quad (17)$$

the kernel k(x) with the Gaussian profile, there is g(x) =-k(x)=1, and equation 17 can be reduced as

$$y_1 = \frac{\sum_{i=1}^{n} x_i w_i}{\sum_{i=1}^{n} w_i} \quad (18)$$

By using equation 18, the mean shift tracking algorithm finds in the new frame the most similar region to the object and it can also be observed that the key parameters in the mean shift tracking algorithm are the weights w i. The obtained new position is checked for threshold and number of iteration before finding it.

1.2.2 Cam Shift Tracking Algorithm

Based on mean-shift, continuously adaptive mean-shift (Cam Shift) was proposed to overcome the problem of mean-shift. Cam Shift adaptively adjusts the track window`s size and the distribution pattern of targets during tracking. Cam Shift algorithm can be used to track the distribution of any kind of feature that represents the target in a lightweight, robust and efficient way.

1) Search Window Initialization

In the majority of works proposed on Cam Shift, the initial location of a target is selected manually by a user. After manually locating the target by a surrounding rectangle, its two dimensional color histogram is calculated for further processing in the next steps.

2) Color Histogram Generation

To obtain robust results, HSV (Hue Saturation Value) color space is used in algorithm for the color histogram generation.

3) Motion Segmentation

To do the motion segmentation in algorithm image differencing method is used which is one of the simplest and most used techniques to detect moving objects. The pixel by pixel intensity difference (D k)of the current frame (f k) and the reference background model (f B) is computed and the resulting image is thresholded to segment the frame`s foreground from its background.

$$D_k(x,y) = |f_k(x,y) - f_B(x,y)| \quad (19)$$

$$Result_k(x,y) = \begin{cases} 0 & \text{background if } D_k \le T \\ 1 & \text{background if } D_k > T \end{cases} \quad (20)$$

The result is a coarse map of temporal changes, after some simple morphological operations on this image, the main moving parts are extracted to generate the image of foreground objects.

4) Probability Distribution

A common method to generate a probability distribution of image is histogram back projection.

5) Mean-shift Application

To calculate the new location of a target, the mean-shift algorithm is used. Mean-shift takes a probability distribution image and an initial search window, computes the window`s center of mass, and then re-centers the window at the computed center of mass. This movement will change what is under the window, and so the re-centering process is repeated until the movement vector converges to zero. The last calculated center of mass will be the new location of the target. The following equations are used to calculate the search window`s center of mass (y_c ,x_c):

$$\begin{cases} x_c = \dfrac{M_{10}}{M_{00}} \\ y_c = \dfrac{M_{01}}{M_{00}} \end{cases} \quad (21)$$

Here the zeroth and first moments are calculated as:

$$\begin{cases} M_{00} = \sum_x \sum_y I(x,y) \\ M_{10} = \sum_x \sum_y xI(x,y) \\ M_{01} = \sum_x \sum_y yI(x,y) \end{cases} \quad (22)$$

Where, I(x,y) is the intensity value of point (x, y) in the probability distribution image. Search window`s new size can be computed as follows:

$$\begin{cases} l = \sqrt{\dfrac{(a+c)+\sqrt{b^2+(a-c)^2}}{2}} \\ w = \sqrt{\dfrac{(a+c)-\sqrt{b^2+(a-c)^2}}{2}} \end{cases} \quad (23)$$

Where, l and w are the long and short axes of the search window respectively, and a, b, and c are obtained from equation (24).

$$\begin{aligned} a &= \frac{M_{20}}{M_{00}} - x_c^2 \\ b &= 2\left(\frac{M_{11}}{M_{00}} - x_c y_c\right) \\ c &= \frac{M_{02}}{M_{00}} - y_c^2 \end{aligned} \quad (24)$$

And the second order moments are calculated from equation (25):

$$\begin{cases} M_{20} = \sum_x \sum_y x^2 I(x,y) \\ M_{02} = \sum_x \sum_y y^2 I(x,y) \\ M_{11} = \sum_x \sum_y xyI(x,y) \end{cases} \quad (25)$$

Finally the new size and location of the search window are used to iterate the algorithm.

1.3 Feature Extraction

1.3.1 Otsu Thresholding and Principal Component Analysis (PCA) Otsu's method is used to automatically perform the reduction of a gray level image to a binary image. The algorithm assumes that the image to be thresholded contains two classes of pixels or bi – modal histogram (e.g. foreground and background) then calculates the optimum threshold separating those two classes so that their combined spread (intra – class variance) is minimal.

"A principal component analysis (PCA) transforms (possibly) correlated variables into uncorrelated variables called principal components by applying an orthogonal transformation, advantage of PCA is that once you have found these patterns in the data, compress the data, that is by reducing the number of dimensions, without much loss of information. This technique used in image compression.

Method

Step: 1 Get some data applied by otsu threshold
Step: 2 Subtract the mean of data
Step: 3 Calculate the covariance matrix
Step: 4 Calculate the Eigen vectors and Eigen value
Step: 5 Choosing components and forming a feature vector
Step: 6 Deriving the new dataset
Step: 7 Getting the old data back

1.4 Feature Detection

1.4.1 Scale Invariant Feature Transform (SIFT) and Speed Up Robust Feature (SURF)

Scale-invariant feature transform (or SIFT) is a robust method proposed by Lowe to find key-points and describe local features. According to Lowe, his descriptors are invariant to image scaling and rotation, and partially invariant to change in illumination and 3D camera viewpoint

The original SIFT algorithm consists of five stages:

1)Scale-space extrema detection

The scale space of an image is defined as a function, L(x, y, σ) that is produced from the convolution of a variable-scale Gaussian, G(x, y, σ), and an input image, I(x, y):

$$L(x,y,\sigma) = G(x,y,\sigma) * I(x,y) \qquad (32)$$

Where, * denotes the convolution operation in x and y, σ represents the scale

$$G(x,y,\sigma) = \frac{1}{2\pi\sigma^2} \exp[-(x^2+y^2)/2\sigma^2] \qquad (33)$$

To efficiently detect stable keypoint locations in the scale-space, difference-of-Gaussian function convolved with the image D(x, y, σ), which can be computed from the difference of two nearby scale

$$D(x,y,\sigma) = \big(G(x,y,k\sigma) - G(x,y,\sigma)\big) * I(x,y) = L(x,y,\sigma) - L(x,y,k\sigma) - L(x,y,\sigma) \qquad (34)$$

Where, k denotes a constant multiplicative factor in two nearby scales. The SIFT algorithm uses $\sigma = 1.6$, $s = 3$, $k = 2^{1/s}$ to construct the scale-space. If the top of the octaves in the scale-space has only 4 pixels, we set,

$$O = \big[\ln(\min(h,w))/\ln(2)\big] - 2 \qquad (35)$$

where w denotes the width of the input image; and h the height of the input image. The scale-space consists of -1,0,1,...,O-1octaves that include 6 Gaussian pyramids with different scales.

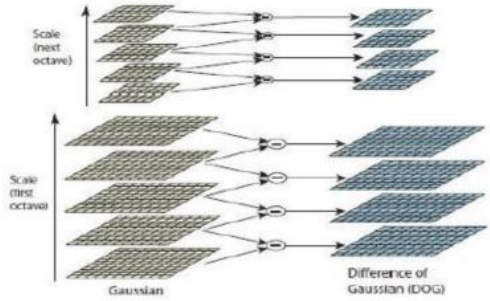


Fig 1: "For each octave of scale space, the initial image is repeatedly convolved with Gaussians to produce the set of scale space images shown on the left. Adjacent Gaussian images are subtracted to produce the difference-of-Gaussian images on the right. After each octave, the Gaussian image is down-sampled by a factor of 2, and the process repeated"

2) Key localization

Identify locations in image scale space that are invariant with respect to image translation, scaling, and rotation, and are minimally affected by noise and small distortions.

As the 2D Gaussian function is separable, its convolution with the input image can be efficiently computed as

$$g(x) = \frac{1}{\sqrt{2\pi\sigma}}\ e^{-x^2/2\sigma^2} \qquad (36)$$

For key localization, all smoothing operations are done using $\sigma = \sqrt{2}$, which can be approximated with sufficient accuracy using a 1D kernel with 7 sample points.

The input image is first convolved with the Gaussian function using $\sigma = \sqrt{2}$ to give an image A. This is then repeated a second time with a further incremental smoothing of $\sigma = (\sqrt{2})$ to give a new image, B, which now has an effective smoothing of $\sigma = (2)$. The difference of Gaussian function is obtained by subtracting image B from A, resulting in a ratio of $2/\sqrt{2} = \sqrt{2}$ between the two Gaussians. Maxima and minima of this scale-space function are determined by comparing each pixel in the pyramid to its neighbors.

3) SIFT key stability

To characterize the image at each key location, the smoothed image A at each level of the pyramid is processed to extract image gradients and orientations. At each pixel, Aij , the image gradient magnitude, Mij , and orientation, Rij, are computed using pixel differences:

$$M_{ij} = \sqrt{(A_{ij} + A_{i+1})^2 + (A_{ij} + A_{i,j+1})^2} \quad (37)$$

$$R_{ij} = \text{atan2}(A_{ij} - A_{i+1,j}, A_{i,j+1} - A_{ij}) \quad (38)$$

4) Local image description

Given a stable location, scale, and orientation for each key, it is now possible to describe the local image region in a manner invariant to these transformations

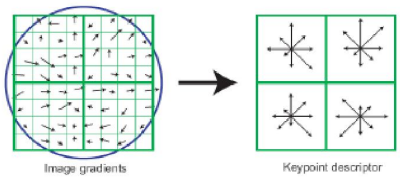


Fig 2: "A key-point descriptor is created by first computing the gradient magnitude and orientation at each image sample point in a region around the keypoint location, as shown on the left. These are weighted by a Gaussian window, indicated by the overlaid circle. These samples are then accumulated into orientation histograms summarizing the contents over 4x4 sub-regions, as shown on the right, with the length of each arrow corresponding to the sum of the gradient magnitudes near that direction within the region. This figure shows a 2x2 descriptor array computed from an 8x8 set of samples, whereas the experiments in this paper use 4x4 descriptors computed from a 16x16 sample array scales, is 8x4x4+8x2x2 or 160 elements, giving enough measurements for high specificity.4x4x8=128 dimensional feature vector".

5) Indexing and matching

For indexing, we need to store the SIFT keys for sample images and then identify matching keys from new images. The best-bin-first search method can identify the nearest neighbors with high probability using only a limited amount of computation.

SURF Algorithm

The Speeded Up Robust Features (SURF) is a high-performance scale and rotation-invariant image key-points detector used to many computer vision tasks such as 3D reconstruction and object recognition. The method was based on some properties of SIFT that uses relative strengths and orientations of gradients to reduce the effect of photometric changes. The idea is to analyze an input image at different scales using Hessian matrices that guarantees scale changes invariance and provide interest points with rotation and scale invariant descriptors. To reduce the time for feature and matching this approach exploit integral images for speed, fast computation of box type convolution filters, distribution of first order Haar wavelet responses in x and y direction rather than the gradient and use only 64 dimensions. According to Bay his methods reduces the time for feature computation and matching, and has proven to simultaneously increase the robustness.

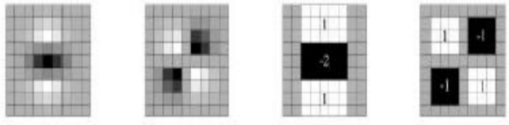


Fig 3: Half left: the (discretised and cropped) Gaussian second order partial derivative; Half right: approximate second order Gaussian derivatives (box filters). The grey regions are equal to zero.

1) Fast-Hessian Detector

Detector is based on the Hessian matrix because of its good performance in computation time and accuracy. Given a point **x** = (x, y) in an image I, the Hessian matrix H(**x**, σ) in **x** at scale σ is defined as follows

$$H(x,\sigma) = \begin{bmatrix} L_{xx}(x,\sigma) & L_{xy}(x,\sigma) \\ L_{xy}(x,\sigma) & L_{yy}(x,\sigma) \end{bmatrix} \quad (39)$$

Where, $L_{xx}(\mathbf{x}, \sigma)$ is the convolution of the Gaussian second order derivative $\frac{\partial^2}{\partial x^2} g(\sigma)$ with the image I in point **x**, and similarly for $L_{xy}(\mathbf{x}, \sigma)$ and $L_{yy}(\mathbf{x}, \sigma)$. Gaussians are optimal for scale-space analysis.In practice, (corresponding to Gaussian derivatives with σ = 1.2).

2) SURF Descriptor

The first step consists of fixing a reproducible orientation based on information from a circular region around the interest point. Then construct a square region aligned to the selected orientation, and SURF descriptor is extracted from it.

3) Orientation Assignment

In order to be invariant to rotation, identify a reproducible orientation for the interest points. For that purpose, first calculate the Haar-wavelet responses in x and y direction and this in a circular neighborhood of radius 6s around the interest point, with s the scale at which the interest point was detected. Also the sampling step is scale dependent and chosen to be s. The side length of the wavelets is 4s.

The horizontal and vertical responses within the window are summed. The two summed responses then yield a new vector. The longest such vector lends its orientation to the interest point.

4) Descriptor Components

For the extraction of the descriptor, the first step consists of constructing a square region centered on the interest point, and oriented along the orientation selected in the previous section. The size of this window is 20s. Each sub-region has a four-dimensional descriptor vector v for its underlying intensity structure v = (dx, dy, |dx|, |dy|), dx is the the Haar wavelet response in horizontal direction and dy the Haar wavelet response in vertical direction (filter size 2s) and sum of the absolute values of the responses, |dx| and |dy|. This results in a descriptor vector for all 4×4 sub-regions of length 64.

1.5 Feature Extraction

1.5.1 Saliency Map

In the area of computer vision saliency detection is important to make one point regarding the type of images considered by the authors of most work in this area. The majority of computer vision image saliency detection work considers high quality and high resolution color images with minimal noise where the salient object is the main subject of the image.

The work that investigated here, with application to saliency detection in UAV images, Intended saliency detection scenario the area on which search and rescue missions are performed can often be narrowed down to a set of fairly uniform environments (e.g. in case of a plane crash on the ocean we would be dealing with a uniform water environment upon which we would be searching for a range of salient debris objects). From this overview of the intended scope of saliency detection for UAV search and rescue (or surveillance) missions the algorithmic architecture of the proposed approach is as follows.

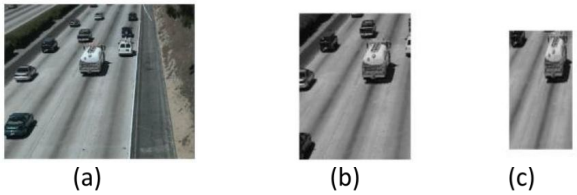
Algorithm Description

A multi-stage salient detection system which combines low-level contrast features, segmentation with additional histogram information and multichannel edge features gathered over several feature maps.

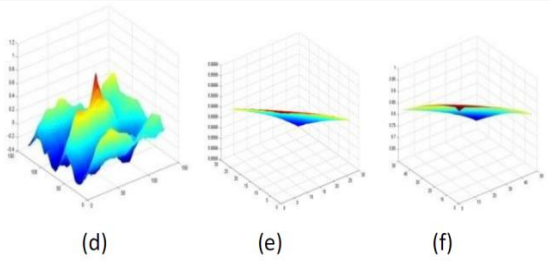
Object Detection

For an original image a minimum scale of gabor filter is taken, difference of image is found by calculating difference between smooth image and grayed image. A multiscale reblurred image required for which Convolution method is used which is more accurate at the edges of the images, where filter kernel is used, from this we get filtered image at edges for input image which uses gabor filter for orientation calculation for an image. From this method a saliency map of original image is found Image Segmentation, Color and Histogram driven Saliency Maps Using segmentation with low-level contrast features has proven to add significant information about the saliency of image regions segmentation on example UAV image. It is important to note that it has left the possible objects of interest as feature regions within the scene. Curve Evolution is done for binary result of Segmentation and for unsupervised method of segmentation for low depth of field by taking exterior points, interior points local and curve evolution uses saliency map, original image and maximum iterations are calculated .

1.6 Experimental Analysis



(a) (b) (c)

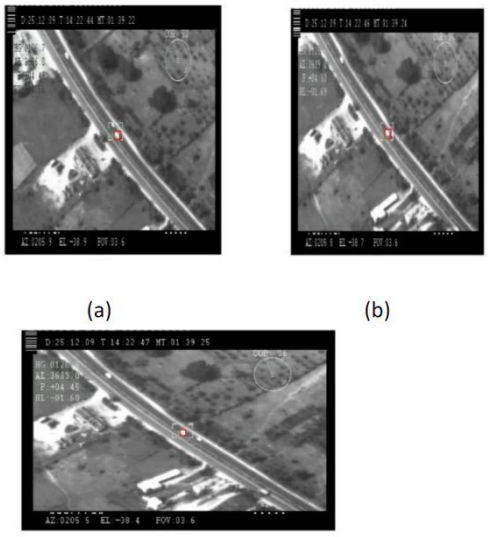


(d) (e) (f)

Fig 4:(a) highway frame. (b) Search Image. (c) Reference Image. (d) 3D surf for Normal – cross correlation for frame(b) and (c). (e) 3D surf for Correlation-1 and Correlation-2 for frame(b) and (c).(f) 3D Brunelli -1 and Brunelli-2 for frame (b) and (c).

Table I: Various Correlation techniques for matching

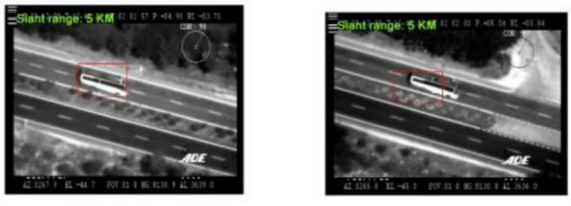
Methods with equation number	Total Computation Time(sec)	Correlation Coefficient
1.Normal-Cross Correlation (4)	2.564110	1
2.Correlation-1(5)	1.697206	0.9999
3.Correlation-2(6)	1.163625	0.9999
4.Brunelli-1 (7)	1.566974	1
5.Brunelli-2 (8)	1.566974	1
6.Khosravi (9)	1.566974	0.1000



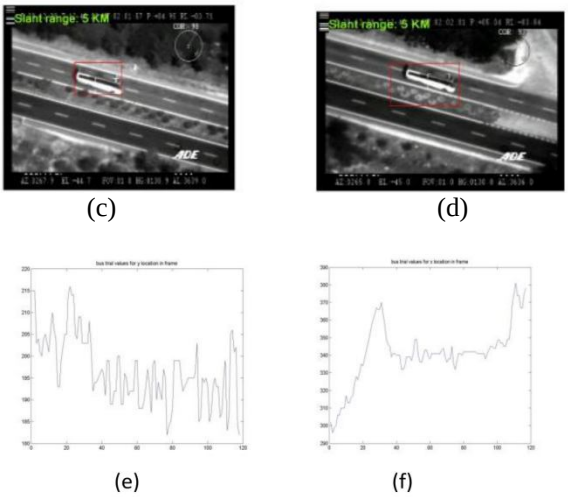
(a) (b)

(c)

Fig 5: (a) Test video frame1(UAV) of video tracking object with window for Mean Shift.(b)Test video frame293 where no occlusion with other car arriving at same position to first car.(c)Test video frame335 where object tracked successfully



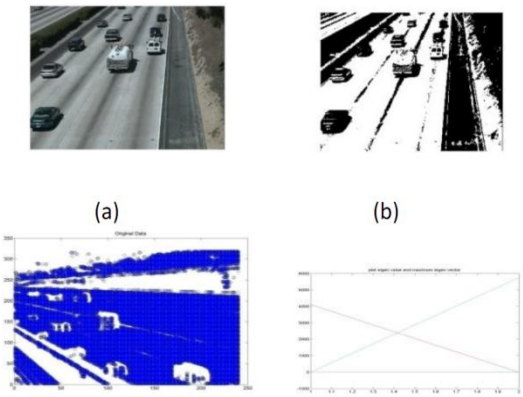
(a) (b)



(c) (d)

(e) (f)

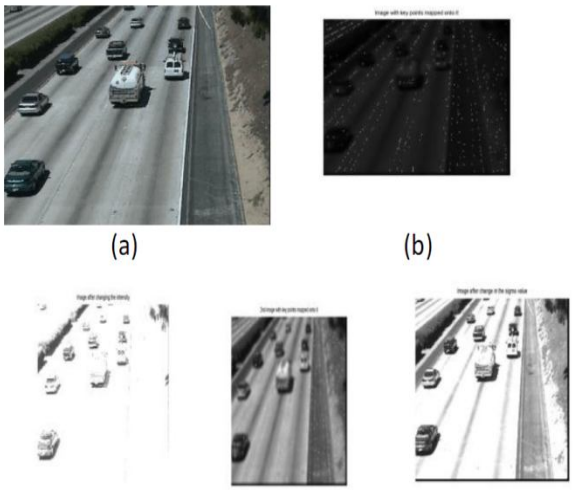
Fig 6:(a)Test video frame1(UAV) of video for Mean shift tracking Bus.(b)Test last video frame118 tracking window misplaced or moved without tracking whole object.(c) Test video frame1(UAV) of video for Cam shift tracking Bus.(d)Test last video frame118 tracking window tracking whole object



(a) (b)

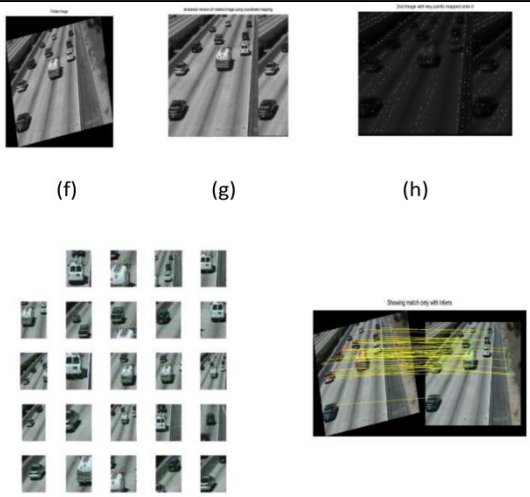
(c) (d)

Fig 7: (a)highway frame.(b)Otsu Thresholded frame with median filter applied.(c)Original datafound by PCA of highway frame for matching.(d)Eigen vector with eigen value merged graph



(a) (b)

(c) (d) (e)



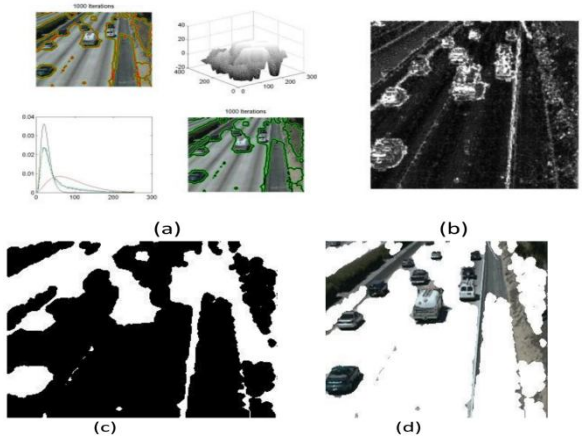
(f) (g) (h)

(i) (j)

Fig 8:(a)highway frame.(b)keypoints of frame(a) by using SIFT method.(c)Intensity increased by 0.2 for frame(a) for matching. (d)Intensity decreased by 0.2 for frame(a) for matching.(e)Scale increased by 0.1 for frame(a) for matching.(f)10° rotation for frame (a)(g)frame(f) is compressed.(h)keypoints found for frame (f)for matching (i)haar features for frame (a).(e)Showing all matches for frame (f) and (a) by SURF method and same match is done for frame (c),(d) and (e).

Table II: Comparison of SIFT and SURF features

Elapsed Time for Computation of key-points and matching(sec)	Increase Intensity(0.2)	Decrease Intensity(0.2)	Scale (0.1)	Rotation(10°)
SIFT	4.6945	4.4612	18.34879	5.086766
SURF	4.73545	4.513263	5.196382	2.439800



(a) (b)

(c) (d)

Fig 9:(a)Object detected for highway frame merged with color histogram driven saliency map.(b)0.5 thresholded frame.(c)Segmentation of highway framewith gabor filter.(d)filtered frame convolution done at edges.

CONCLUSION

Feature based Image Matching for Digital Scene Matching Area Correlation (DSMAC), Image Registration, Automatic Acquisition can be done using different techniques. This framework is robust for scale changes like rotation and translation,

Intensity change like increase intensity, decrease in intensity and scale change for images in gray scale and colored images. Compared to all techniques SIFT and SURF technique is efficient for scale invariance required for matching by using key points. Real time application of for matching is the feature work which should show robustness for scale invariance for UAV.

REFERENCES

[1]. Yeonha Hwang, Min – Jea Tahk "Terrain Referenced UAV Navigation with LIDAR – A Comparison of Sequentail Processing and Batch Processing Algorithms" ICAS 2012, 28th INTERNATIONAL CONGRESS OF THE AERONAUTICAL SCIENCES

[2]. Chapter 5-"Trace Scheduling based Optimization of Navigation System Component"

[3]. Gerald Jean Francis Banon and Sergio Donizette Faria. "Area Based Image Matching Algorithm Assessment from Satellite Images" Image Anal Stereol 2001;20(Suppl 1):573-578 Proc 8thECS and Image Analysis, September 4-7, 2001, Bordeaux, France.

[4] R.Collins, "Mean-shift Tracking", Spring 2006.

[5]. Ebrahim Emami, Mahmood Fathy "Object Tracking Using Improved CAMShift Algorithm Combined with Motion Segmentation" 2011 IEEE.

[6]. David G. Lowe "Object Recognition from Local Scale-Invariant Features" Proc. of the International Conference on Computer Vision, Corfu (Sept. 1999).

[7]. Bay,H,. Tuytelaars, T., &Van Gool, L.(2006). "SURF: Speeded Up Robust Features", 9th European Conference on Computer Vision.

[8]. Ricardo C. Bonfim Rodrigues and Sergio Roberto M. Pellegrino, "An Experimental Evaluation of Algorithms for Aerial Image Matching" IWSSIP 2010 - 17th International Conference on Systems, Signals and Image Processing (SIFT, SURF)

[9]. Luo Juan & Oubong Gwun, "A Comparison of SIFT, PCA-SIFT and SURF" International Journal of Image Processing (IJIP) Volume (3), Issue (4)

[10]. Jan Sokalski, Toby P. Breckon, "AUTOMATIC SALIENT OBJECT DETECTION IN UAV IMAGERY", 25th International UAV Systems Conference.

★★★