Introduction

Today, Visual Tracking is more important due to its wide applications in intelligent visual surveillance, human–computer interaction, augmented reality, driver assistance, robot vision, and so on. In any kind of field, to discover the suitable solution from available solutions, one should use optimization techniques (Yilmaz et al., 2006). Now to find the best suitable solution, mathematical optimization which is a constraint based process is proposed. Video tracking is the process of locating a moving object or multiple objects over time using a camera. It has a variety of uses, Some of which are: human-computer interaction, security and surveillance, video communication and compression, augmented reality, traffic control, medical imaging http://en.wikipedia.org/wiki/Video_tracking - cite_note-1 and video editing. Video tracking can be a time consuming process due to the amount of data that is contained in video. Adding further to the complexity is the possible need to use object recognition techniques for tracking, a challenging problem in its own right. The objective of video tracking is to associate target objects in consecutive video frames. The association can be especially difficult when the objects are moving fast relative to the frame rate. Another situation
that increases the complexity of the problem is when the tracked object changes orientation over time. For these situations video tracking systems usually employ a motion model which describes how the image of the target might change for different possible motions of the object.

In traditional approach, particle filtering which relies only on random sampling for state optimization is used. The key idea of particle filtering is to represent the required posterior density function by a set of random samples with associated weights. Though particle filtering has a lower probability to be trapped in local maxima, the optimal importance function for sampling is often not available, so usually a very large number of particles (drawn from the prior dynamic model) are needed to approximate the posterior density. An ISPF framework is implemented for visual tracking on the affine group, which can find the optimal state in a chain like way with a very small number of particles proposed by Li et al., (2010, 2012). ISPF uses an online-learned pose estimator to guide random particles to move toward to their neighboring best states with the help of learned pose estimation, random particles become smart and sparse (thin) sampling becomes possible (Babenko et al., 2009; Doucet et al., 2000). Particles can be incrementally drawn from a motion prior and then can be tuned iteratively toward the neighborhood of the optimal state by the pose estimation. The result is that a set of particles forms a short chain in the state space and efficiently finds the optimal state. Sampling is terminated if the maximum similarity of all tuned particles satisfies a target-patch similarity distribution modeled online or if the permitted maximum number of particles is reached. With the help of the learned PE and some appearance-similarity feedback scores, particles in ISPF become “smart” and can automatically move toward the correct directions; thus, sparse sampling is possible (Li et al., 2012). The optimal state can be efficiently found in a step-by-step way in which some particles serve as bridge nodes to help others to reach the optimal state. In addition to the single-target scenario, the “smart” particle idea is also extended into a multi target tracking problem. This framework demonstrates that the ISPF can achieve great robustness and very high accuracy with only a very small number of particles.

**Multivariate normal distribution**

Multivariate normal distribution is a probability distribution in a multivariate analysis. Multivariate normal distribution has a mean \( \mu \) and variance-covariance matrix \( \Sigma \) of random n-vector \( X \) and is denoted as \( X \sim \mathcal{N}(\mu, \Sigma) \) and its density is given by equation—(1)

\[
\frac{1}{(2\pi)^{n/2}|\Sigma|^{1/2}} \exp\left(\frac{1}{2}(x-\mu)^T \Sigma^{-1} (x-\mu)\right)
\]

The following is the very special property of multivariate normal distribution, which is used to test the independency of the random variable.

**Independence test for poses**

Let \( X \) is a normal random vector. The components are independent if they are uncorrelated. i.e., \( \text{Cov}(X_i, X_j) = 0 \) then they are uncorrelated so the two components \( X_i \) and \( X_j \) are independent (Mishra et al., 2012).

In this paper we used this property in the following two cases:

**Case 1:** We have to compare all shapes of images and check whether all belong to one input image or not. In this case if they are not uncorrelated then all shape of images belongs to one particular image. i.e., \( \text{Cov}(X_i, X_j) \neq 0 \) and \( X_i, X_j \in C \) (\( X_i, X_j \) are from
poses of images) which means they are not independent, which also implies that there is some relation between these poses. The example for covariance matrix has been shown in Figure 1.

**Case 2:** After succession of step 1, from all the angles of images we have to test which shape is the best match to the input images. In this case, we have to test the independency property for the input image and the shapes of images i.e., Cov(X_i, X_j) ≠ 0. Here if we find any one of the shape is not independent to the input image, it is regarded as the target inference for the input image.

**Fig.1** Illustration of motion tracking. The transpose matrix are the homogeneous coordinates of pixels in object regions at Frame #0 and Frame #t, respectively; R_t is the transformation to be obtained by the tracking process.

**Proposed algorithm for part based ISPF**

1. Initially, Take the video file converts into Frame Set
2. Select the frame from frame set
3. Take difference between them using Bayesian filter estimates of the covariance and Correlated
4. If both image frame covariance and correlation is unbiased then
   - Display “Both are same”
   - Else
     - Display “Not the Same”
5. End

**Least square estimation of part based images**

Parameter estimation plays a center off attraction for software reliable approximation. This approach of reliable likelihood commonly contains two different ways as follows; one is to estimate the parameters that the input data is directly taken into equations. The other approach is fitting the curve described by the function to the data and estimating the parameters from the best fit to the curve. The most common method for this indirect parameter estimation is the least squares technique. In this, we estimate the value of one variable with the value of the other known variable. The statistical method which helps us to estimate the unknown value of one variable from the known value of the related variable is called regression (Gupta and Kapoor, 2009; Zivkovic et al., 2006). The least squared data is as shown in Table 1 and Table 2. In this approach, there are two methods for studying regression namely, Graphic and Algebraic methods. In this section, we study the image recognition failure data sets using an algebraic method called as least square estimation. It indicates the best possible mean value of one variable corresponding to the mean value of the other. Here, we can compute the pose data set coefficients of the equation Y=a+bX by solving the normal equation.

Regression equation of y on x is given is equations (2) and (3)

\[ \sum y = b \sum x + Na \]  
\[ \sum xy = b \sum x^2 + a \sum x \]

\[ (2) \]

\[ (3) \]
Part based image data for least square estimation

<table>
<thead>
<tr>
<th>S.No.</th>
<th>Input Image (X)</th>
<th>Failure Data Output (Y)</th>
<th>X²</th>
<th>X.Y</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.0445</td>
<td>2</td>
<td>1.09098</td>
<td>2.089</td>
</tr>
<tr>
<td>2</td>
<td>1.0536</td>
<td>3</td>
<td>1.110073</td>
<td>3.1608</td>
</tr>
<tr>
<td>3</td>
<td>1.0659</td>
<td>6</td>
<td>1.136143</td>
<td>6.3954</td>
</tr>
<tr>
<td>4</td>
<td>1.0839</td>
<td>11</td>
<td>1.174839</td>
<td>11.9229</td>
</tr>
<tr>
<td>5</td>
<td>1.0876</td>
<td>12</td>
<td>1.182874</td>
<td>13.0512</td>
</tr>
<tr>
<td>6</td>
<td>1.093</td>
<td>16</td>
<td>1.194649</td>
<td>17.488</td>
</tr>
<tr>
<td>7</td>
<td>1.1168</td>
<td>19</td>
<td>1.247242</td>
<td>21.2192</td>
</tr>
<tr>
<td>8</td>
<td>1.1422</td>
<td>19</td>
<td>1.304621</td>
<td>21.7018</td>
</tr>
<tr>
<td>9</td>
<td>1.1517</td>
<td>20</td>
<td>1.326413</td>
<td>23.034</td>
</tr>
<tr>
<td>10</td>
<td>1.1901</td>
<td>22</td>
<td>1.416338</td>
<td>26.182</td>
</tr>
<tr>
<td>11</td>
<td>1.2191</td>
<td>23</td>
<td>1.486205</td>
<td>28.0393</td>
</tr>
<tr>
<td>12</td>
<td>1.225</td>
<td>25</td>
<td>1.500625</td>
<td>30.625</td>
</tr>
</tbody>
</table>

Table I Pose image data

<table>
<thead>
<tr>
<th>S.No.</th>
<th>Least Square Parameter Estimation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>a=15.327873, b=-0.4404565</td>
</tr>
</tbody>
</table>

Table II Part based image data least square estimation

Experimental Results

Analysis of Variance (ANOVA) is a hypothesis-testing technique used to test the equality of two or more population means by examining the variances of samples that are taken. ANOVA allows one to determine whether the differences between the samples are simply due to random error (sampling errors) or whether there are systematic treatment effects that cause the mean in one group to differ from the mean in another. The test images are shown in Figure 2 and Figure 3 respectively.

Solution: The null hypothesis for an ANOVA always assumes the population means are equal. Hence, we may write the null hypothesis as: the Null Hypothesis is H₀: All the means values of are not same. Since the null hypothesis assumes all the means are equal, we could reject the null hypothesis if only mean is not equal. Thus, the alternative hypothesis is: Hₐ: At least one mean pressure is statistically equal. Hypothesis H₀ at 1% level and conclude that All the means values are same.

![Fig.2 Part based pose oriented frames](image1)

![Fig.3 Part based pose oriented image frames](image2)

![Table III: ANOVA Table](image3)
Conclusion

An incremental self-tuning particle filtering (ISPF) framework is implemented for pose tracking on the same person groups, which can find the optimal state in a chain like way with a very small number of part based particles. Before going to test the poses using Independent test conducted with the help of multivariate normal distribution is used. Finally, ANOVA test conducted on part based pose estimation in same person frames.

References


