

# Real time object tracking to remove occlusion using OpenCV

Aniruddh Thakor, Anjali Askhedkar

**Abstract**— Real time Object tracking is becoming a challenging ingredient in analysis of video imagery for efficient and robust object tracking. In this paper we presents a how to remove occlusion problem from real time video, removing occlusion from video is still a challenging part. Object tracking with sparse Prototypes, exploits both classic Principal Component Analysis (PCA) algorithms and sparse representation algorithms for learning appearance models. Here, regularization into the PCA reconstruction is introduced and this algorithm to represent an object by sparse prototypes that account precisely for data and noise is developed. In order to reduce tracking drift, a method that considers occlusion and motion blur into account rather than simply image observations for model update is given. The proposed tracking algorithm performs favorably against various methods that can be demonstrated by both qualitative and quantitative estimations on challenging image sequences.

**Index Terms**— Object tracking, sparse prototype, Principle component analysis (PCA), L1 minimization, incremental visual tracking (IVT) and Compressive tracking

## 1 INTRODUCTION

As one of the problems in computer vision, online object tracking plays a critical role in research such as motion analysis, image compression and activity recognition. Difficulties including intrinsic and extrinsic factors to account for the appearance changes of a target object affects the development of robust online tracker as in past decades most of the progress have already been made.

A tracking method typically consists of three components Observation model, Dynamic model, Search strategy.

The observation model is concerned with how objects are represented. Any representation scheme can be categorized based on adopted features like intensity, color, texture, Haar-like feature, etc and description models like holistic histogram, part-based histogram, and subspace representation. Instead of treating the target object as a collection of low-level features, subspace representation methods provide a compact notion of the “thing” being tracked, which facilitates other vision tasks.

Dynamic model is describes the states of an object over time. For object tracking, generative methods focus on modeling appearance and formulate the problem as finding the image observation with minimal reconstruction error and the other side, discriminative algorithms aims to determine a decision boundary that distinguishes the target from the background.

Search strategy is for finding the states in the current

- Aniruddh Thakor is currently pursuing Masters of Engineering in Electronics and Telecommunication in MIT College of Engineering, University of Pune, India, PH-07304055333. E-mail: ani.thakor@gmail.com
- Anjali Askhedkar is currently Assistant Professor in Electronics and Telecommunication in MIT College Of Engineering, University of Pune, India, PH-09850826663. E-mail: anjali.askhedkar@mitcoe.edu.in

frame and sliding window. Online learning facilitates tracking algorithms by adapting to appearance changes of the target and the background. It includes various methods like template update, incremental subspace learning and online classifiers have been demonstrated to be effective for object tracking.

Sparse coding algorithms [1] is modeling data vectors as sparse linear combinations of basic elements is generally used in machine learning, neuroscience, signal processing, and statistics. The recent development of sparse coding representation has been proved to be very effective for signal reconstruction and classification in the audio and image processing domains (image denoising, image classification, and object tracking, etc). These methods have been proved that a learning dictionary from data outperforms pre-chosen ones since the former can significantly reduce reconstruction error. Different from the representations based on PCA and its variants, sparse models do not impose that the bases in the dictionary be orthogonal, which allows more flexibility to adapt the representation to the data. A robust generative tracking algorithm with adaptive appearance model which handles partial occlusion and other challenging factors is proposed in this project. By exploiting the advantage of subspace representation, this algorithm is able to process higher resolution image observations, and performs more efficiently with favorable results than the existing method based on sparse representation of templates.

## 2 RELATED WORK

These techniques are based purely on 2D techniques for recognition, image registration and handling of illumination changes. This fact instantly differentiates the approach from systems that either attempt to estimate a 3D model from 2D input or require a 3D data as an input. Although these techniques can give better robustness to pose variation given an exact 3D model, for access control applications where only moderate pose variation is present, the proposed method will be more sufficient. Note that 2D images of faces under changing brightness already contain 3D shape-related information and this information can be leveraged by 2D algorithms for recogni-

tion and alignment even if shape is not reconstructed explicitly.

Correspondence between points in the training and the test image should be achieved in case of holistic recognition algorithms. Presently research exist on Active Appearance Models [10], and the closely related Active Shape Model [11] to register images against a relatively high-dimensional model of conceivable face appearances, often leveraging face specific contours. The advantages of this model are variations in expression and pose, they add complexity to applications where subjects normally present a neutral face or have moderate expression. Here we have some freedom (i.e. similarity transformations, etc). Iterative registration in this spirit dates at least back to the Lucas-Kanade algorithm [12].

Early work on image registration is aimed at the problem of registering nearly identical images; say by maximizing normalized correlation or minimizing a sum of squared distances, here we must confront several physical factors simultaneously: illumination variations, misalignment, and corrupted pixels. Illumination variation can be taken care by expressing the test image as a linear combination of an appropriate set of training images. Illumination robust tracking shows similar representations. For gross to robustness errors, the L2-norm of the residual is a less appropriate objective function than the classical L1-norm. Its use here is less motivated by theoretical results due to Cande's and Tao [10]. These two observations lead us to pose the registration problem as illumination coefficients that minimize the L1-norm of the representation error and the search for a set of transformations. All of these problems can also be solved efficiently using first-order techniques for L1-minimization [5].

### 3 OBJECT TRACKING WITH INCREMENTAL SUBSPACE LEARNING & SPARSE REPRESENTATION

Object tracking via online subspace learning has been engaging much attention in recent years. The incremental visual tracking (IVT) method [7] introduces an online update approach for efficient learning and updating a low dimensional PCA subspace representation of the target object. Various experimental results demonstrate that PCA subspace representation using online update is effective in dealing with appearance changes caused by in-plane rotation, scale, illumination variation and pose change. Though, it has also been shown that the PCA subspace based representation scheme is sensitive to partial occlusion.

$$y = Uz + e \quad (1)$$

Where  $y$  denotes an observation vector,  $z$  indicates the corresponding coding or coefficient vector,  $U$  represents a matrix of column basis vectors, and  $e$  is the error term.

In PCA, the underlying assumption is that the error vector  $e$  is Gaussian distributed with small variances (i.e., small dense noise). Therefore, the coding vector  $z$  can be described by  $z = U^T y$ . But, this assumption does not hold

for object representation in visual tracking when partial occlusion occurs as the noise term cannot be modeled with small variance. Hence, the IVT method is sensitive to partial occlusion. In addition, the IVT method is not equipped with an effective update mechanism since it simply uses new observations for Learning new basis vectors without detecting partial occlusion and processing these samples accordingly.

Sparse representation has been largely studied and applied in pattern recognition and computer vision (face recognition, super resolution, and image in painting, etc). An algorithm is proposed by casting the tracking problem as handling partial occlusion with trivial templates and by finding the most likely patch with sparse representation.

$$y = Az + e = [A|I] \begin{bmatrix} z \\ e \end{bmatrix} = Bc \quad (2)$$

Where,  $y$  denotes an observation vector,  $A$  represents a matrix of templates,  $z$  indicates the corresponding coefficients, and  $e$  is the error term which can be viewed as the coefficients of trivial templates.

By assuming that each candidate image patch is sparsely represented by a set of target and trivial templates, Eq. 2 can be solved via L1 minimization [5].

$$\min \frac{1}{2} \|y - Bc\|_2^2 + \lambda \|c\|_1 \quad (3)$$

The underlying assumption of this approach is that error  $e$  can be modeled by arbitrary but sparse noise, and therefore it can be used to handle partial occlusion. However, the  $l_1$  tracker has two main drawbacks. First, the computational complexity limits its performance. As it requires solving a series of  $l_1$  minimization problems, it often deals with low resolution images as a tradeoff of speed and accuracy. Such low-resolution images may not capture sufficient visual information to represent objects for tracking. The  $l_1$  tracker is computationally expensive even with further improvements. Second, it does not exploit rich and redundant image properties which can be captured compactly with subspace representations. We present an efficient and effective representation those factors out the part describing the object appearance and the other part for noise.

### 4 COMPRESSIVE TRACKING

Tracking algorithms can be generally categorized as either generative or discriminative based on their appearance models. Generative tracking algorithms typically learn a model to represent the target object and then use it to search for the image region with minimal reconstruction error. Discriminative algorithms pose the tracking problem as a binary classification task in order to find the decision boundary for separating the target object from the background.

Compressive tracking [2] is an effective and efficient tracking algorithm with an appearance model based on features extracted in the compressed domain. The main components of compressive tracking algorithm are

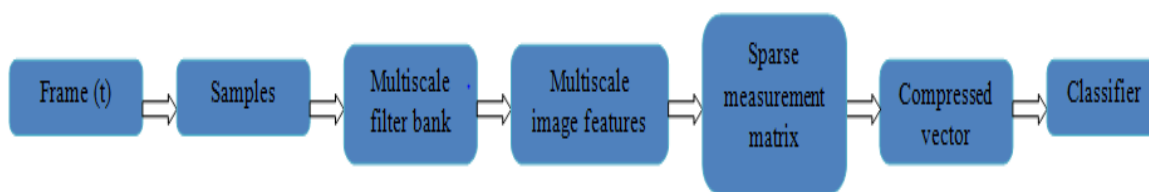


Fig. 1 Updating Classifier at the t-th frame

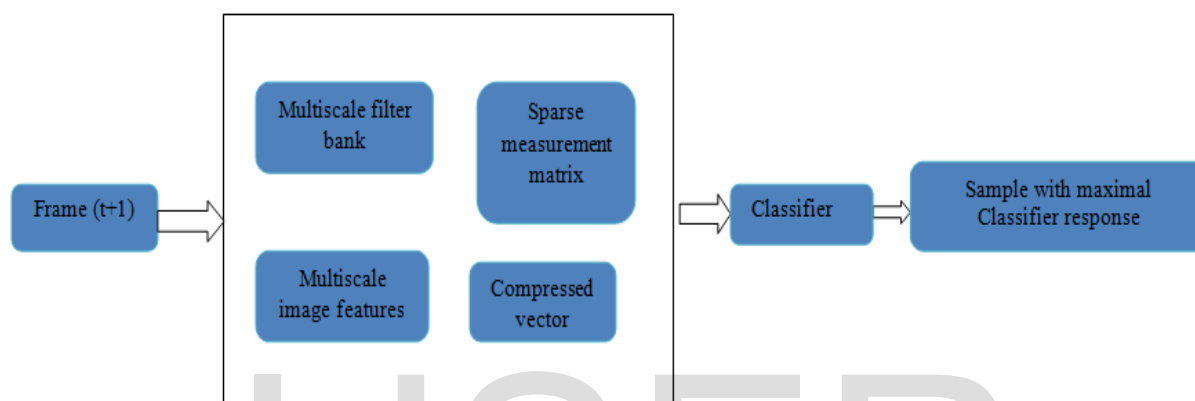


Fig. 2 Updating classifier at t+1 th frame

object can be well represented based on the features extracted in the compressive domain. It is also discriminative because we use these features to separate the target from the surrounding background via a naive Bayes classifier.

The tracking problem is formulated as a detection task and our algorithm is shown in Fig 1& Fig 2. We assume that the tracking window in the first frame has been determined. At each frame, we sample some positive samples near the current target location and negative samples far away from the object center to update the classifier. To predict the object location in the next frame, we draw some samples around the current target location and determine the one with the maximal classification score.

#### 4.1 Compressive tracking Algorithm

1. Sample a set of image patches, where  $l$   $t-1$  is the tracking location at the  $(t-1)$  -th frame, and extract the features with low dimensionality.
2. Use classifier to each feature vector and find the tracking location  $l_t$  with the maximal classifier response
3. Sample two sets of image patches
4. Extract the features with these two sets of samples and update the classifier parameters

shown in Fig. 1. Appearance model is generative as the **5 IMPLEMENTATION**

For implementation purpose, we used OpenCV platform has been used. OpenCV has advantages over existing platforms such as high processing speed (approximately 25-30 frame per second), open source, library of programming functions mainly aimed at real time computer vision, originally written in C but now has a full C++ interface. Because of high processing speed it is suitable for real time applications.

We have used OpenCV2.3 and Microsoft visual studio 2008 as a debugger. Language used for coding is C++.

#### 5 RESULTS

The results of proposed tracking methods are shown below. Fig.3 shows results for steady state object. Fig.4 shows optical flows when occlusion occurs and fig.5 shows results when object is partially occluded. The bounding box shows the target object. The results are taken on the basis of compressive tracking.

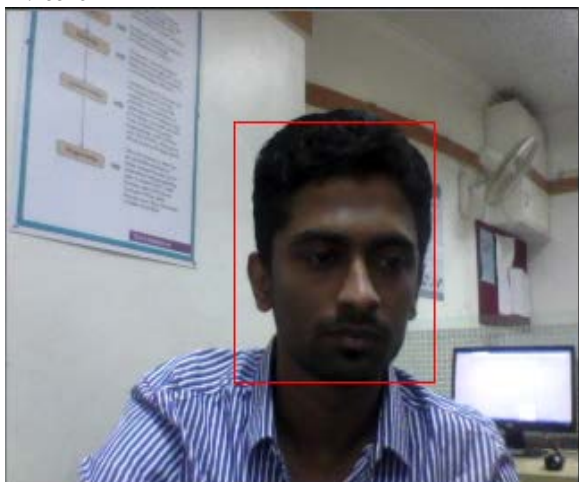


Fig. 3 Compressive tracking when object is steady

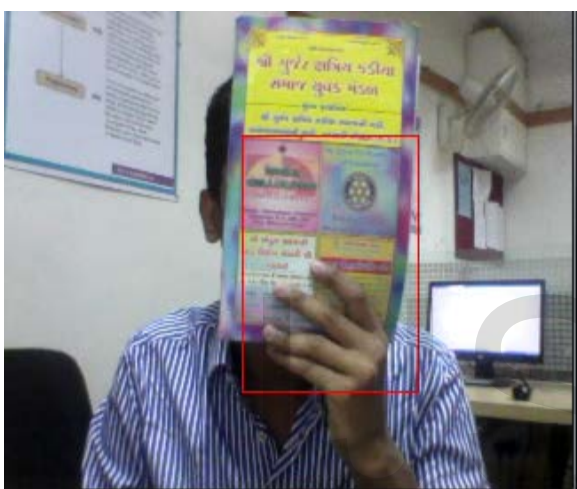


Fig. 4 Compressive tracking when object is fully occluded

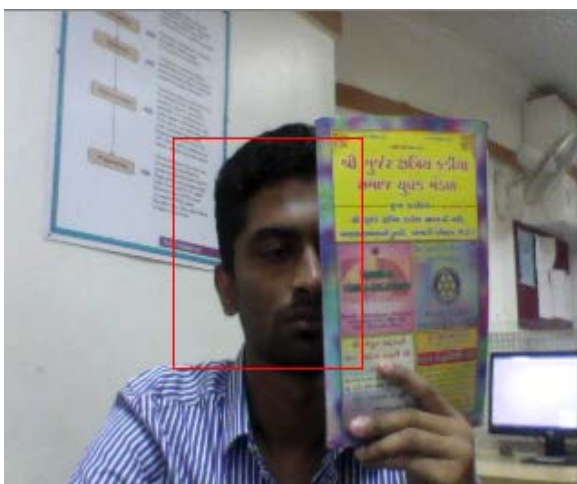


Fig. 5 Compressive tracking when object is partially occluded

occlusion. The system achieves extremely stable performance under a wide range of variations in illumination, misalignment, and even under small amounts of pose and occlusion.

## REFERENCES

- [1]. Dong Wang, Huchuan Lu, Member, IEEE, and Ming-Hsuan Yang, Senior Member, IEEE, "Online Object Tracking With Sparse Prototypes", Vol.22, NO.1, Jan 2013.
- [2]. Kaihua Zhang<sup>1</sup>, Lei Zhang<sup>1</sup>, and Ming-Hsuan Yang<sup>2</sup>. Real-Time Compressive Tracking. A. Fitzgibbon et al. (Eds.): ECCV 2012, Part III, LNCS 7574, pp. 866–879, 2012
- [3]. D. Wang, H. Lu, and Y.-W. Chen, "Incremental MPCA for color object tracking," in Proc. IEEE Int. Conf. Pattern Recogn., Aug. 2010.-
- [4]. Z. Kalal, J. Matas, and K. Mikolajczyk, "P-N learning: Bootstrapping binary classifiers by structural constraints," in Proc. IEEE Conf. Computer Vision Pattern Recogn., Jun. 2010, pp. 49–56.
- [5]. J. Kwon and K. M. Lee, "Visual tracking decomposition," in Proc. IEEE Conf. Comput. Vision Pattern Recogn., Jun. 2010, pp. 1269–1276.
- [6]. X. Mei and H. Ling, "Robust visual tracking using L1 minimization," in Proc. IEEE Int. Conf. Comput. Vision, Sep.–Oct. 2009, pp. 1436–1443.
- [7]. D. Ross, J. Lim, R.-S. Lin and M. H. Yang, "Incremental learning for robust visual tracking," Int. J. Comput. Vision, vol. 77, nos. 1–3, pp. 125–141, 2008.
- [8]. H. Grabner and H. Bischof, "On-line boosting and vision," in Proc. IEEE Conf. Comput. Vision Pattern Recogn., Jun. 2006, pp. 260–267.
- [9]. E. Candes and T. Tao, "Decoding by linear programming," IEEE Transactions on Information Theory, vol. 51, no. 12, 2005.
- [10]. D. Comaniciu, V. R. Member, and P. Meer, "Kernelbased object tracking," IEEE Trans. Pattern Anal. Mach. Intel., vol. 25, no. 5, pp. 564–575, May 2003 1751–1754.
- [11]. T. Cootes, G. Edwards, and C. Taylor, "Active appearance models," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 23, no. 6, pp. 681–685, 2001
- [12]. T. Cootes and C. Taylor, "Active shape models – 'smart snakes'," in Proceedings of British Machine Vision Conference, 1992.
- [13]. B. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in Proceedings of International Joint Conference on Artificial Intelligence, vol. 3, 1981, pp. 674

## 6 CONCLUSION

This paper presents a visual object tracking algorithm via the proposed sparse prototype representation. In this paper we explicitly take partial occlusion and motion blur into account for appearance update and object tracking by exploiting the strength of subspace model and sparse representation. We propose sparse representation for robust visual tracking in real time environment using OpenCV. We used compressive tracking for removing

# IJSER