Integrated Hidden Markov Model and Kalman Filter for Online Object Tracking

C. V. Sakthi Priya[1]
[1]PGP College Of Engineering And Technology, Namakkal

Abstract— Visual prior from generic real-world images study to represent that objects in a scene. The existing work presented online tracking algorithm to transfers visual prior learned offline for online object tracking. To learn complete dictionary to represent visual prior with collection of real world images. Prior knowledge of objects is generic and training image set does not contain any observation of target object. Transfer learned visual prior to construct object representation using Sparse coding and Multiscale max pooling. Linear classifier is learned online to distinguish target from background and also to identify target and background appearance variations over time. Tracking is carried out within Bayesian inference framework and learned classifier is used to construct observation model. Particle filter is used to estimate the tracking result sequentially however, unable to work efficiently in noisy scenes. Time sift variance were not appropriated to track target object with observer value to prior information of object structure. Proposal HMM based kalman filter to improve online target tracking in noisy sequential image frames. The covariance vector is measured to identify noisy scenes. Discrete time steps are evaluated for identifying target object with background separation. Experiment conducted on challenging sequences of scene. To evaluate the performance of object tracking algorithm in terms of tracking success rate, Centre location error, Number of scenes, Learning object sizes, and Latency for tracking.

Keywords: Visual prior, object tracking, object recognition, sparse coding, and transfer learning.

I. INTRODUCTION

Online object tracking is a challenging problem as it entails learning an effective model to account for appearance change caused by intrinsic and extrinsic factors. In this paper, we propose a novel online object tracking algorithm with sparse prototypes, which exploits both classic principal component analysis (PCA) algorithms with recent sparse representation schemes for learning effective appearance models. We introduce `1 regularization into the PCA reconstruction, and develop a novel algorithm to represent an object by sparse prototypes that account explicitly for data and noise. For tracking, objects are represented by the sparse prototypes learned online with update.

The goal of object tracking is to estimate the locations and motion parameters of a target in an image sequence given the initialized position in the first frame. Research in tracking plays a key role in understanding motion and structure of objects. It finds numerous applications including surveillance, human computer interaction, track pattern analysis, recognition, medical image processing, to name a few. Although object tracking has been studied for several decades, and numerous tracking algorithms have been proposed for different tasks, it remains a very challenging problem. There exists no single tracking method that can be successfully applied to all tasks and situations.

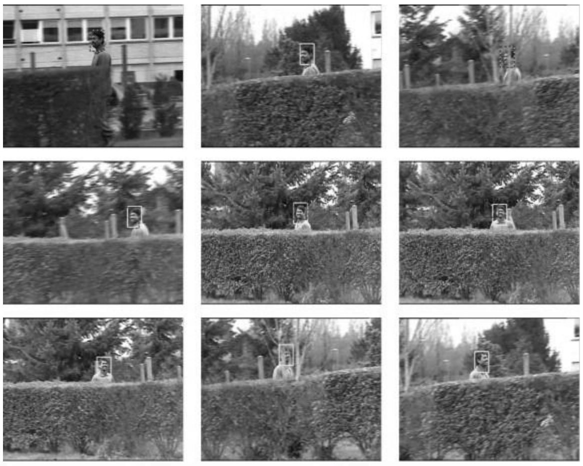

Fig. 1: Online Object Tracking

Therefore, it is crucial to review recent tracking methods, and evaluate their performances to show how novel algorithms can be designed for handling specific tracking scenarios

Visual prior from generic real-world images can be learned and transferred for representing objects in a scene. Motivated by this, we propose an algorithm that transfers visual prior learned offline for online object tracking. From a collection of real-world images, we learn an over complete dictionary to represent visual prior. The prior knowledge of objects is generic, and the training image set does not necessarily contain any observation of the target object. During the tracking process, the learned visual prior is transferred to construct an object representation by sparse coding and multiscale max pooling. With this representation, a linear classifier is learned online to distinguish the target from the background and to account for the target and background appearance variations over time.

Tracking is then carried out within a Bayesian inference framework, in which the learned classifier is used to construct the observation model and a particle filter is used to estimate the tracking result sequentially. Experiments on a variety of challenging sequences with comparisons to several state-of-the-art methods demonstrate that more robust object tracking can be achieved by transferring visual prior.

The use of hidden Markov models for speech recognition has become predominant in the last many years, as evidenced by the number of published papers and talks at major speech conferences. The reasons this method has become so popular are the inherent statistical (Mathematic

-ally precise) framework; the ease and availability of training algorithms for estimating the parameters of the models from finite training sets of speech data; the flexibility of the resulting recognition system in which one can easily change the size, type, or architecture of the models to suit particular words, sounds, and so forth; and the ease of implementation of the overall recognition system. In this expository article, we address the role of statistical methods in this powerful technology as applied to speech recognition and discuss a range of theoretical and practical issues that are as yet unsolved in terms of their importance and their effect on performance for different system implementations.

The Kalman filter, also known as linear quadratic estimation (LQE), is an algorithm that uses a series of measurements observed over time, containing noise (random variations) and other inaccuracies, and produces estimates of unknown variables that tend to be more precise than those based on a single measurement alone. More formally, the Kalman filter operates recursively on streams of noisy input data to produce a statistically optimal estimate of the underlying system state. The filter is named for Rudolf (Rudy) E. Kalman, one of the primary developers of its theory.

II. LITERATURE SURVEY

In computer vision, the data are rarely independent since their labelling is related due to spatiotemporal dependencies. The data with dependent labels will be called structured. For instance, in object detection, the task is to label all possible image patches of an input image either as positive (object) or as negative (background) [1]. Sparse coding is an unsupervised algorithm that learns to represent input data succinctly using only a small number of bases. For example, using the “image edge” bases, it represents a new image patch as a linear combination of just a small number of these bases. Informally, we think of this as finding a representation of an image patch in terms of the “edges” in the image; this gives a slightly higher-level/more abstract representation of the image than the pixel intensity values, and is useful for a variety of tasks [2].

Low dimensional eigenspace representation is learned online, and is updated incrementally over the time. The framework only assumes that the initialization of the object region is provided. Second, while the Condensation algorithm is used for propagating the sample distributions over the time, we develop an effective probabilistic likelihood function based on the learned tensor eigenspace model. Third, while R-SVD is applied to update both the sample mean and eigenbasis online as new data arrive, an incremental multilinear subspace analysis is enabled to capture the appearance characteristics of the object during the tracking [3]. Invariant local feature matching could be extended to general image recognition problems in which a feature was matched against a large database of images. They also used Harris corners to select interest points, but rather than matching with a correlation window, they used a rotationally invariant descriptor of the local image region [4].

Working with noisy images recorded by digital cameras is difficult since different devices produce different kinds of noise, and introduce different types of artefacts and spatial correlations in the noise as a result of internal post-processing (demosaicking, white balance, etc.). Non-local means filtering has proven very effective in general, but it fails in some cases [5]. Patches can be sampled densely, randomly, or at selected salient points. Various local descriptors exist with different degrees of geometric and photometric invariance, but all encode the local patch appearance as a numerical vector and the more discriminant ones tend to be high-dimensional. The usual way to handle the resulting set of descriptor vectors is to vector quantizes them to produce so-called textons or visual words [6].

The static part in our system is based on normalized cross-correlation (NCC). We simply use the object which is marked in the first frame by a rectangle as template and match it in every forthcoming frame. The tracking rectangle is moved to the peak of the correlation confidence map. NCC does not adapt to any changes but brightness, which renders it useless when the object appearance changes permanently [7]. The architecture performs only two major kinds of computations (template matching and max pooling) while some of the other systems involve complex computations like the estimation of probability distributions or the selection of facial components for use by an SVM. Perhaps part of the model’s strength comes from its built-in gradual shift- and scale-tolerance that closely mimics visual cortical processing, which has been finely tuned by evolution over thousands of years [8]. The bag-of-features (BoF) model has been extremely popular in image categorization.

III. INTEGRATED HIDDEN MARKOV MODEL AND DALMAN FILTER FOR ONLINE OBJECT TRACKING

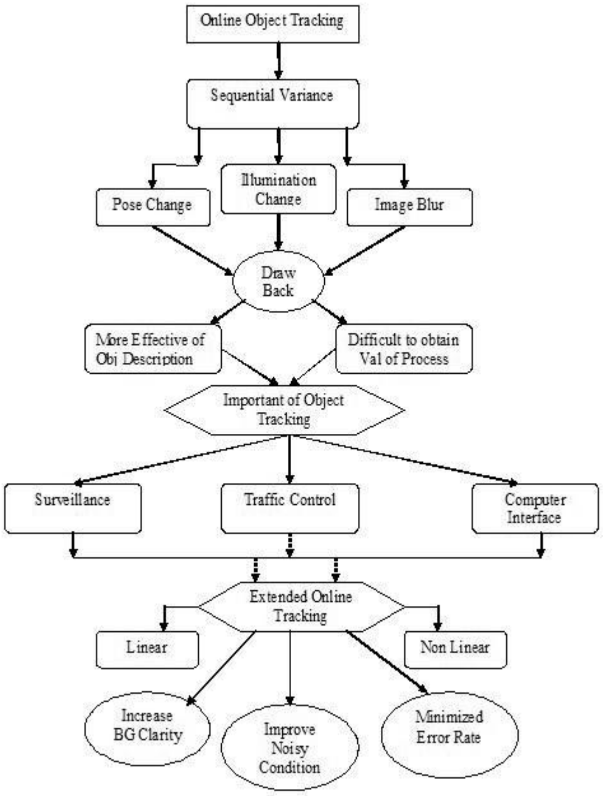

Fig. 2: Architecture Diagram

The method treats an image as a collection of unordered appearance descriptors extracted from local patches, quantizes them into discrete “visual words”, and then computes a compact histogram representation for semantic image classification.

The BoF approach discards the spatial order of local descriptors, which severely limits the descriptive power of the image representation. By overcoming this problem, one particular extension of the BoF model, called spatial pyramid matching (SPM), has made a remarkable success on a range of image classification benchmarks. Sparse modelling of image patches has been successfully applied to tasks such as image and video de-noising, inpainting, de-mosaicing, super-resolution and segmentation [9]. On the other hand, if multiple positive examples are used (taken from a small neighbourhood around the current tracker location), the model can become confused and its discriminative power can suffer. Alternatively, a semi-supervised approach where labelled examples come from the first frame only, and subsequent training examples are left unlabelled [10].

The phases involved in the proposed schemes are

1) Learn visual Prior
2) Sparse Coding and Multi-scale
3) Linear Classifier and online tracking
4) Kalman Filter and noisy reduction
5) Discrete time steps with HMM

A. *Learn visual prior*

Learn visual prior from large image sets of diverse objects with over complete dictionary to utilize sparse coding. On patch level the small images often share structural similarity that is why we exploit such prior. The exploit prior information offline from existing data sets are used for online visual tracking. Two data sets contain large variety of objects are used for learning visual prior. The use of object classes, including non-rigid (face, person and dog), rigid (bicycle, bus, car and motorbike) objects and other related object images are used to learn a prior for specific tracking tasks. Dictionary is learned offline with sparse coding method. The dictionary contains generic structured information of different objects from numerous classes that are used to encode generic visual prior of objects. It learned dictionary contains structural information of object appearance that represent object with less reconstruction error

B. *Sparse Coding and Multi-scale*

Sparse coding based on SIFT descriptor is used as basic appearance descriptor in the tracking method. It is used to extract SIFT descriptors from overlapped patches of each grayscale image. Learn the dictionary in an unsupervised manner and visual prior is represented by over complete dictionary. Transfer prior for object tracking by representing an object. For each SIFT descriptor inside an object region that sparse coefficient vector is learned. Object is represented by applying multi-scale max pooling on coding results of all local SIFT descriptors in corresponding image region. For tracking task is define object level feature for a target or background sample over sparse representation matrix. Use pooling function that operates on each row of matrix and obtain a vector. Use max pooling function on the Absolute Sparse codes for robust local spatial translations. It preserves spatial information and local invariance, use multi-scale max pooling to obtain object level representation of pooling process searches across different. Pooling process searches across different locations over different scales of the object image and combines all local maximum responses. Divide whole object image into non overlapped spatial cells apply max pooling on coding results of descriptors in each cell that concatenate the pooled features from all spatial cells.

C. *Linear Classifier and online tracking*

Linear classifier is learned to separate the target from the background use logistic regression to learn the classifier and classification score as our similarity measure for object matching, pose tracking is binary classification. Patches from different objects in the same category are represented by similar sparse coefficients that object level representation based on multi-scale max pooling exploits the unique spatial characteristics of a target object. Representation scheme consists of local feature response and geometric shape. Learned classifier is target-specific used to discriminate a target from objects in the same or different categories. Classifier is updated with most recent observations and is updated with recently obtained target observations and negative samples extracted in the current frame. Only most recently obtained target observations are stored in the memory. Negative samples are collected in the current image frame that are need minimum large memory requirement and flexible to deal with long sequences.

D. *Kalman Filter and noisy reduction*

Parameters are necessary for tracking are x and y coordinates for u and v velocity components. Nodes are represented by vector that observed position of the target in the context of visual tracking. It describe state through a finite set of parameters is state-space model. State nodes are related to each other through underlying object motion. The transition from one state to the next is described in many ways. Different alternatives can be grouped into linear and non-linear functions describing state transition. Standard Kalman filter employs a linear transition function. The extended Kalman filter (EKF) allows a non-linear transition together with a non-linear measurement relationship. Kalman filter model requires representation of the tracker in state space. Noise error covariance information is estimated from analysis of the process. Two parameters required in each state are x and y coordinates of the tracker. State transition matrix is derived from theory of motion under constant acceleration. State includes velocity components and acceleration components.

E. *Discrete time steps with HMM*

Data on which Kalman filter operates is represented with HMM is based on the idea of modelling sequential data. Individual image objects were modelled as a sequence of interdependent nodes. Dependencies between image object were discovered through statistical analysis of object nature. In the context of visual tracking sequence is made up of images taken at discrete time steps. Relationship between each of the images is based on a physical model of the scene.

IV. EXPERIMENTAL RESULTS AND DISCUSSIONS

In this section we evaluate performance of offline visual prior information for online object tracking through Java implementation. One of the major contributions of this work is the design of an online object tracking with offline visual prior information based on the Integrated Hidden Markov Model and Kalman Filter for Online Object Tracking. To confirm the analytical results, we implemented Integrated Hidden Markov Model and Kalman Filter for Online Object Tracking in the implementation of Java and evaluated the performance of services.

The performance of Integrated Hidden Markov Model and Kalman Filter for Online Object Tracking is evaluated by the following metrics.

1) Tracking Success Rate
2) Centre Location Error
3) Latency for Tracking

Number of Scenes	Existing Transferring Visual Prior	Proposed Integrates Hidden Markov Model and Kalman Filter
1	120	150
2	86	130
3	75	110
4	63	100
5	50	70

Table 1: Tracking Success Rate

Figure 3 demonstrates the tracking success rate. X axis represents the number of scenes whereas Y axis denotes the tracking success rate using both the Transferring Visual Prior our proposed Integrated Hidden Markov Model and Kalman Filter.

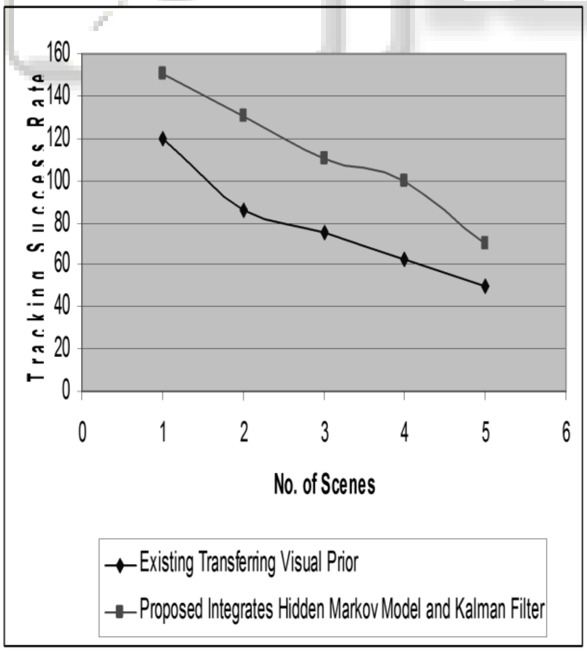

Fig. 3: Tracking Success Rate

When the number of scenes increased number of tracking success rate also gets increased. All the curves show a more of less yet steady descendant when scenes increase. Figure 2 shows better estimates for tracking success rate of Integrated Hidden Markov Model and Kalman Filter. Integrated Hidden Markov Model and Kalman Filter achieve 15% to 23% more tracking success rate result.

Number of Scenes	Existing Transferring Visual Prior	Proposed Integrates Hidden Markov Model and Kalman Filter
1	3.6	1.2
2	4.7	1.9
3	5.1	2.5
4	6.9	3.4
5	7.2	4.5

Table 2: Centre Location Error

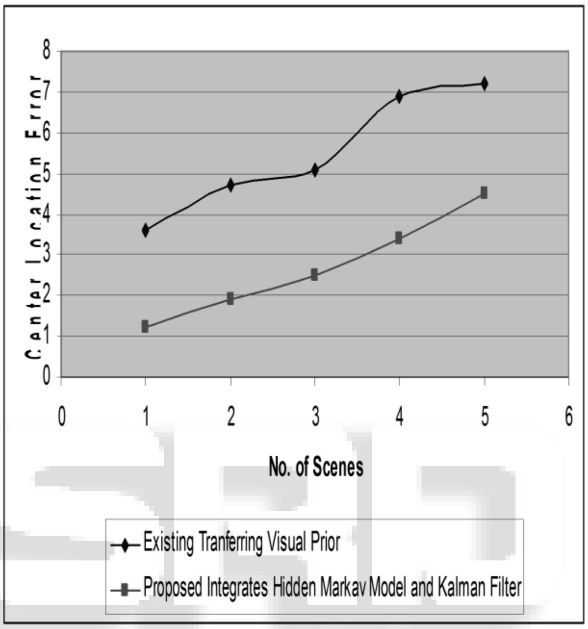

Fig. 4: Centre Location Error

Number of Scenes	Existing Transferring Visual Prior	Proposed Integrates Hidden Markov Model and Kalman Filter
1	250	60
2	290	110
3	342	130
4	342	130
5	450	230

Figure 4 demonstrates the centre location rate. X axis represents number of scenes whereas Y axis denotes the centre location error using both the existing Transferring Visual Prior our proposed Integrated Hidden Markov Model and Kalman Filter.

When the number of scenes increased centre location error Also gets increased. Figure 3 shows the effectiveness of centre location rate over different number of scenes than Transferring Visual Prior our proposed Integrated Hidden Markov Model and Kalman Filter. Integrated Hidden Markov Model and Kalman Filter achieve 20% to 30% less centre location rate when compared with existing schemes.

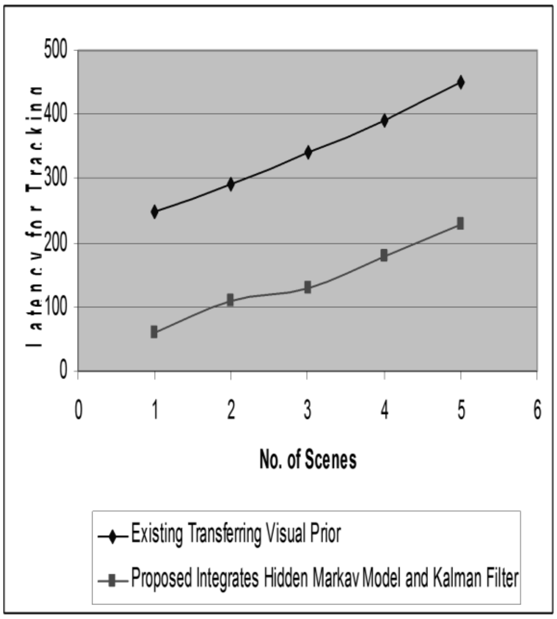

Fig. 5: Latency for Tracking

Figure 5 demonstrates the latency for tracking. X axis represents the number of scenes whereas Y axis denotes the latency for tracking using both the Transferring Visual Prior our proposed Integrates Hidden Markav Model and Kalman Filter. When the number of scenes increased, latency for tracking also gets increases accordingly. The latency for tracking is illustrated using the existing Transferring Visual Prior our proposed Integrates Hidden Markav Model and Kalman Filter. Figure 4 shows better performance of Proposed Integrates Hidden Markav Model and Kalman Filter in terms of scenes than existing Transferring Visual Prior our proposed Integrates Hidden Markav Model and Kalman Filter. Integrates Hidden Markav Model and Kalman Filter achieve 40 to 65% less latency for tracking variation when compared with existing system.

V. CONCLUSION

This work exploits generic visual prior learned from real world images for online tracking of specific objects. On the patch level, small images often share structural similarity, and such prior information can be learned offline and used for modelling objects in online visual tracking. We have presented an effective method that learns and transfers visual prior for robust object tracking. With a large set of natural images, we represent visual prior with an over-complete dictionary. We transfer the learned prior to tracking tasks by sparse coding and represent the object with the multi-scale max pooling method. Develop an online object tracking with offline visual prior information for given specific image sequences through Integrated Hidden Markov Model and Kalman Filter for Online Object Tracking. With newly arrived samples of the target and background, a classifier is learned online to discriminate the target object from background and a particle filter algorithm is utilized to estimate the target state sequentially. The implementation is used to improve object tracking in noisy conditions. The measurements of target object location is more refined and prior knowledge of object localization is made more specific.

REFERENCES

[1] Z. Kalal, J. Matas, and K. Mikolajczyk. P-n learning: Bootstrapping binary classifiers by structural constraints. In Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pages 49–56, 2010.

[2] H. Lee, A. Battle, R. Raina, and A. Ng. Efficient sparse coding algorithms. In Advances in Neural Information Processing Systems, 2007.

[3] X. Li and W. Hu. Robust visual tracking based on incremental tensor subspace learning. In Proceedings of IEEE International Conference on Computer Vision, pages 1–8, 2007.

[4] D. Lowe. Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision, 60(2):91–110, 2004.

[5] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman. Nonlocal sparse models for image restoration. In Proceedings of IEEE International Conference on Computer Vision, pages 2272–2279, 2009.

[6] F. Moosmann, B. Triggs, and F. Jurie. Fast discriminative visual codebooks using randomized clustering forests. In Advances in Neural Information Processing Systems, pages 985–992, 2006.

[7] J. Santner, C. Leistner, A. Saffari, T. Pock, and H. Bischof. Prost: Parallel robust online simple tracking. In Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pages 723– 730, 2010.

[8] T. Serre, L. Wolf, and T. Poggio. Object recognition with features inspired by visual cortex. In Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, volume 2, pages 994–1000, 2005.

[9] J. Yang, K. Yu, Y. Gong, and T. Huang. Linear spatial pyramid matching using sparse coding for image classification. In Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pages 1794– 1801, 2009.

[10] B. Babenko, M.-H. Yang, and S. Belongie. Visual tracking with online multiple instance learning. In Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pages 983–990, 2009.