A HIERARCHICAL FEATURE MODEL FOR MULTI-TARGET TRACKING

Mohib Ullah[1] *Ahmed Kedir Mohammed*[1] *Faouzi Alaya Cheikh*[1] *and Zhaohui Wang*[2]

[1]Norwegian University of Science and Technology, Gjøvik, Norway
[2]Hainan University, China

ABSTRACT

We propose a novel Hierarchical Feature Model (HFM) for multi-target tracking. The traditional tracking algorithms use handcrafted features that cannot track targets accurately when the target model changes due to articulation, illumination intensity variation or perspective distortions. Our HFM explore deep features to model the appearance of targets. Then, we use an unsupervised dimensionality reduction for sparse representation of the feature vectors to cope with the time-critical nature of the tracking problem. Subsequently, a Bayesian filter is adopted as the tracker and a discrete combinatorial optimization is considered for target association. We compare our proposed HFM against 4 state-of-the-art trackers using 4 benchmark datasets. The experimental results show that our HFM outperforms all the state-of-the-art methods in terms of both Multi Object Tracking Accuracy (MOTA) and Multi Object Tracking Precision (MOTP).

***Index Terms*—** Hierarchical Feature Model, multi-target tracking, deep features, sparse representation, Bayesian filter, combinatorial optimization.

1. INTRODUCTION

In computer vision, visual target tracking is an active and challenging topic with applications including but not limited to surveillance, activity analysis, autonomous vehicles, smart drones. Huge effort is dedicated to this problem from the computer vision community and many breakthroughs have been achieved in the past few years. Nevertheless, many challenges still exist in conceiving a tracking framework that can cope illumination variation, background clutter and most importantly, target model variations. Therefore, it is still an open research problem and there is room for improvement

Modeling the appearance of a target is one of the most important building blocks of any tracking framework. Until now, tracking algorithms relied on handcrafted features [1–7]. The common approach in these techniques is to characterize each target with a handcrafted feature descriptor and then find the similar targets in the consecutive frames through a similarity matrix like JS divergence [6].

Handcrafted features are statically obtained by a predefined rule. They can be classified into two groups i.e. local and global. Local handcrafted features characterize a specific region in an image. Some of the popular local handcrafted features are BRIEF [8], and SIFT [9]. Global handcrafted features characterize the whole image rather than a specific region in it. Commonly used global handcrafted features are HoG [10], color histogram, and LBP [11].

Over the past few years, techniques based on deep learning have become the methods of choice for traditional computer vision problems such as image classification, object detection and speech recognition. By far, the most famous and well studied supervised deep learning algorithm is Convolutional Neural Network (CNN). Different architectures of CNN were proposed in recent years and they have been the winners [12–14] of the Imagenet challenge for object recognition since 2012.

CNN is inspired from natural vision systems and were introduced over two decades ago. However, due to lack of computing power and unavailability of large collections of labeled data, it could not be exploited to its full potential. After the exponential growth of portable electronic devices, the amount of annotated data increased rapidly which lead to large datasets such as Imagenet [15]. Likewise, high performance GPUs solved the problem of network training through parallelism. As a result, CNN became the preferred algorithm for object recognition and classification tasks. However, they are not exploited for tracking which leaves much potential for research. In this paper, we try to explore this direction and exploit deep features in a Bayesian framework for multi-target tracking. The contributions of this work are two-folds:

- We propose a novel Hierarchical Feature Model (HFM). To the best of our knowledge, this is the first approach where deep features learned from Imagenet dataset, are explored to model the appearance of the targets for multi-target tracking. Moreover, we analyzed different layers of a CNN and found their imperative properties for target tracking. We observed that features extracted from the lower layers of a CNN are generic and better suited for a task like target tracking.
- Second, we employ an unsupervised dimensionality reduction for the sparse representation of the feature vectors. As a result, an overall processing boost is achieved in the tracking framework.

978-1-5090-2175-8/17/$31.00 ©2017 IEEE

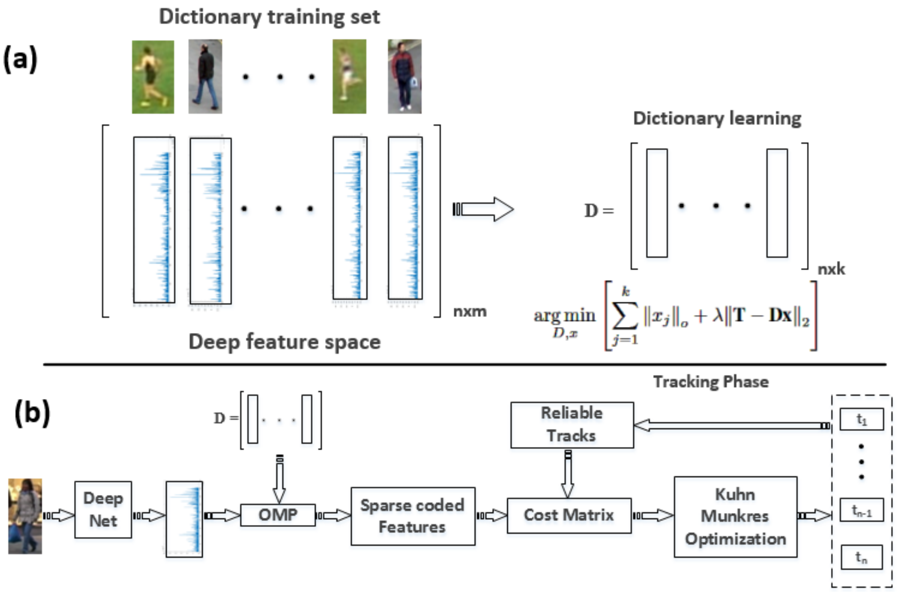

Fig. 1: (a) Targets are chosen in the first 100 frames of each dataset randomly. Selected target's feature vectors are used for dictionary learning. (b) Feature vector of a target is sparse coded using the learnt dictionary and OMP algorithm. It is compared with all the targets in the previous frame and correct association is established through Kuhn Munkres algorithm. Afterwards, it is used as the observation model of the corresponding Bayesian filter.

The rest of the paper is organized in the following way: In section 2, a brief overview of the proposed approach is given. In section 3, Bayesian filtering is explained. Deep feature based appearance model is explained in section 4. In section 5, dimensionality reduction is illustrated. In section 6, quantitative results of experiments are given and section 7 concludes the paper.

2. PROPOSED HFM APPROACH

In this section a brief review of each step of the proposed HFM approach is given. The block diagram is given in Fig. 1. We focus on pedestrians as the targets of interest. Initially, a dictionary is learnt from a collection of frames taken from the datasets. The non-convex optimization problem of finding the most sparse representation of a feature vector is handled through Orthogonal Matching Pursuit (OMP) [16]. We adopted Google's inception architecture of CNN [13] as the feature extractor. However, rather than using all the layers for feature extraction, we used only the lower 7 layers. It is because lower layers learn generic features while higher layers learn class specific features [17]. In our analysis, we found that only lower layers are useful because the targets are very generic. The sparse coded feature vectors model the appearance of the targets which is treated as the observation model of the Bayesian filter. Target association is addressed as a discrete combinatorial optimization problem and Kuhn Munkres algorithm is employed for associating the targets in the consecutive frames. In the following sub-sections, each part of the algorithm is explained.

3. BAYESIAN FILTERING

Bayesian filtering is a probabilistic approach for estimating an unknown probability distribution recursively through sequential observations and a dynamic system model. In the context of target tracking, the objective is to approximate the state $\mathbf{s}$ of a target from the noisy observations $\mathbf{z}$. Where the state $\mathbf{s}$ represents the position and velocity of a target. Using the Markovian property, the probability distribution of a state $\mathbf{s}$ at time t given all the observation $\mathbf{z}_{1:t}$ is approximated as

$$p(\mathbf{s}_t|\mathbf{z}_{1:t}) = p(\mathbf{z}_t|\mathbf{s}_t) \int_{\mathbf{s}_{t-1}} p(\mathbf{s}_t|\mathbf{s}_{t-1})p(\mathbf{s}_{t-1}|\mathbf{z}_{1:t-1})d\mathbf{s}_{t-1} \tag{1}$$

where $p(\mathbf{s}_t|\mathbf{s}_{t-1})$ and $p(\mathbf{z}_t|\mathbf{s}_t)$ show the state transition probability and the observation likelihood at time t, respectively. Generally, Eq. 1 is evaluated in two steps: a prediction step which uses a dynamic system model to predict the state of a target and an update step which corrects the prediction. By imposing the constraint of linearity and gaussianity for the state and observation model, an exact solution of Eq. 1 is given by Kalman filter [18]. In our HFM approach, we instantiated one Kalman filter per target in the scene.

In the case of a single target, Bayesian filtering is straight forward. However, in multi-target tracking, an additional step is needed to find which observation is associated with which target. Our proposed HFM approach models the appearance of targets through deep features and uses a discrete combinatorial optimization approach for solving the association problem. We adopted Kuhn Munkres [19] algorithm which follows a greedy approach to associate the targets in two consecutive frames before the update step of the Bayesian filter.

4. TARGET APPEARANCE MODEL

In a nutshell, appearance of a target can be characterized by two types of features i.e. handcrafted and learned features. Compared to handcrafted features, deep features are not extracted by a predefined rule but they are learned from a large dataset through a learning process. A cost function is used to optimize the learning process and back-propagation [20] is used as baseline learning algorithm. The architecture of a CNN is trained end-to-end and the filters which are responsible for extracting deep features are learned in the training process. Intrinsically, deep features are used by a CNN for image classification. However, the whole network can be partitioned into two functional blocks i.e. feature extraction block and the classification block. The work of Sharif et al. [21] have shown that if a CNN is trained on a large scale dataset and used as a generic feature extractor, it still gives better performance than handcrafted features for image classification.

Inspired from this, we explored deep features to model the appearance of the targets for multi-target tracking. We truncated the full network into its functional blocks and used the first functional block as the feature extractor. However, rather than using all the layers of the network for feature extraction, we used lower layers to extract only generic features. For the sparse representation of the features, an unsupervised method is adopted to learn over complete set of bases which is explained in the next section.

5. DIMENSIONALITY REDUCTION

In the class of unsupervised methods, sparse coding is a technique for characterizing data optimally. Compared to Principal Component Analysis (PCA) which learns complete set of base vectors, sparse coding learns over-complete set of base vectors. An over-complete set of basis helps to represent the structures and patterns in the input data.

The analysis and visualization of deep features [17] show that different portions of the feature vector share similar attributes including color and texture. Similar attributes can be exploited to represent the feature vector efficiently through sparse coding. Sparse signal representation has become very popular in the past few years and lead to state-of-the-art results in various applications such as face recognition [22] and image denoising [23]. The main goal of sparse modeling is to efficiently represent the images as a linear combination of a few typical patterns, called atoms, which are selected from a dictionary. Here, we intend to exploit sparse representation of the deep features to reduce the processing time of the overall tracking framework.

5.1. Sparse Coding

Sparse learning aims at finding a sparse representation of the input data in the form of a linear combination of the atoms. Given a dictionary matrix, $\mathbf{D} \in \mathbb{R}^{n\times k}$ that contains k atoms as column vectors $\mathbf{d}_j \in \mathbb{R}^n, j = 1, 2, ..., k$, the sparse coding problem of a signal $\mathbf{y} \in \mathbb{R}^n$ can be stated as finding the sparsest vector $\mathbf{x} = [x_1, x_2, ..., x_k] \in \mathbb{R}^k$ such that

$$\mathbf{y} \approx \sum_{j=1}^{k} x_j d_j \tag{2}$$

or the representation error $\mathbf{R} = \mathbf{y} - \mathbf{D}\mathbf{x}$ is minimized, therefore the optimization problem can be formulated as

$$\arg\min_x \|\mathbf{x}\|_o, \tag{3}$$

subjected to

$$\|\mathbf{y} - \mathbf{D}\mathbf{x}\|_2 \leq \varepsilon, \tag{4}$$

where ε is the reconstruction error of the signal $\mathbf{y}$ using the dictionary $\mathbf{D}$ and sparse code $\mathbf{x}$. Alternatively, the optimization problem can be formulated as

$$\arg\min_x \sum_{j=1}^{k} \|x_j\|_o + \lambda \|\mathbf{y} - \mathbf{D}\mathbf{x}\|_2, \tag{5}$$

with a regularizer λ. The minimization problem above is not convex because of the ℓ_0 norm and solving this problem is NP-hard [24]. Hence, there are approximate solutions using greedy approaches such as Orthogonal Matching Pursuit (OMP) [16]. For this work, we adopted OMP method for sparse representation [25, 26].

5.2. Dictionary Learning

A dictionary is a collection of a key feature patterns known as atoms. A common setup for the dictionary learning problem starts with access to a training vector $\mathbf{T} = [t_1, ..., t_m]$, where each $t_i \in \mathbb{R}^m$. We obtain the training set from the dataset which we used in experiment. K-SVD (K-Singular Value Decomposition) is used to iteratively solve the optimization problem of Eq. 6, by alternatively computing the sparse approximation of $\mathbf{x}$ using OMP and then the algorithm proceeds to update the atoms of the dictionary $\mathbf{D}$.

$$\arg\min_{D,x} \left[\sum_{j=1}^{k} \|x_j\|_o + \lambda \|\mathbf{T} - \mathbf{D}\mathbf{x}\|_2 \right] \tag{6}$$

K-SVD [27] is an iterative method that alternates between sparse coding of the training set based on the current dictionary and a process of updating the dictionary atoms to better fit the data.

6. EXPERIMENTS

The proposed HFM is implemented in Matlab with the support of MatConvNet toolbox. The processing is done on Intel core i7 with 8 GB RAM. The performance of our HFM is compared against 4 state-of-the-art methods on 4 benchmark datasets and results show a substantial improvement beyond the state-of-the-art. Quantitative results are shown on MOTA and MOTP [28] which are the standard performance matrices for multi-target tracking algorithms. All the datasets are recorded outdoor in an unconstrained environment and have strong variability in resolution, video quality, and frame rate. Reference methods which are used for the comparison are using hand-crafted features. The quantitative results from Table 1 show that our method gives superior performance in terms of both MOTA and MOTP. For the dictionary learning, we used only 10 frames from each dataset (40 in total) which are chosen from the first 100 frames randomly. It is important to note that in the first 100 frames of each dataset, there was no error in tracking. Almost all the errors occur at the end of each dataset. It can be seen in Fig. 2 (f) where the bounding box of a target drifted. It is because the dictionary is learnt only with 40 frames. However, if more frames are used for training the dictionary, better results can be achieved. However, it would take more time to learn the dictionary. The complete qualitative results are also attached in the supplementary material which are obtained with our experimental setup.

Datasets	Methods	MOTA	MOTP
Pets2009	Berclaz et al. [2]	82.0%	56.0%
	Dehghan et al. [4]	90.4%	63.12%
	Andriyenko et al. [5]	89.3%	56.4%
	Proposed HFM	94.8%	74.6%
TUD-crossing	Dehghan et al. [3]	91.9%	70.0%
	Dehghan et al. [4]	92.9%	69.2%
	Proposed HFM	93.0%	72.8%
TUD-Stadmitt	Berclaz et al. [2]	45.8%	73.9%
	Dehghan et al. [3]	82.4%	73.9%
	Andriyenko et al. [5]	61.8%	63.2%
	Proposed HFM	89.1%	84.5%
AFL1	Proposed HFM	92.3%	87.7%

Table 1: Quantitative results of our HFM approach. The results show that our method outperform traditional methods both on MOTA and MOTP.

7. CONCLUSION

We explored deep features to model the appearance of the targets for multi-target tracking. Features are extracted only from the lower layers of a CNN and an unsupervised dimensionality reduction approach is employed for the sparse representation of the feature vector which results in a processing boost in the overall tracking framework. Quantitative results show that deep features give superior performance than hand-crafted features.

In future, we are aiming to extend this approach to multiple overlapping cameras and exploit the redundant information for multi-target tracking. Moreover, we are also aiming to use transfer learning and train only higher layers of the CNN on a short tracking dataset while keep the pretrained lower layers intact.

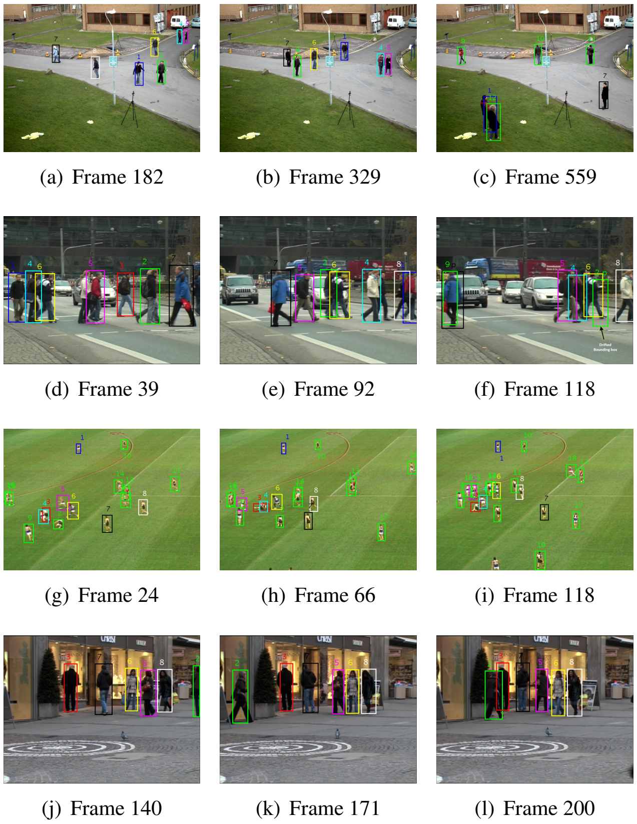
(a) Frame 182 (b) Frame 329 (c) Frame 559
(d) Frame 39 (e) Frame 92 (f) Frame 118
(g) Frame 24 (h) Frame 66 (i) Frame 118
(j) Frame 140 (k) Frame 171 (l) Frame 200

Fig. 2: Tracking results obtained with our HFM approach. Top to bottom: datasets Pets2009 (a-c), TUD-crossing (d-f), AFL (g-i) and TUD-Stadtmitte (j-l). Targets are given a unique integer ID.

Acknowledgments

This research was funded by Norwegian Ministry of education and research and Hainan provincial scientific collaboration project under grant KJHZ2015-23.

8. REFERENCES

[1] Jing Wang, Hong Zhu, Shunyuan Yu, and Caixia Fan, "Object tracking using color-feature guided network generalization and tailored feature fusion," *Neurocomputing*, vol. 238, pp. 387–398, 2017.

[2] Jerome Berclaz, Francois Fleuret, Engin Türetken, and Pascal Fua, "Multiple object tracking using k-shortest paths optimization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 9, pp. 1806–1819, 2011.

[3] Afshin Dehghan, Shayan Modiri Assari, and Mubarak Shah, "Gmmcp tracker: Globally optimal generalized maximum multi clique problem for multiple object tracking," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 4091–4099.

[4] Afshin Dehghan, Yicong Tian, Philip HS Torr, and Mubarak Shah, "Target identity-aware network flow for online multiple target tracking," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 1146–1154.

[5] Anton Andriyenko, Konrad Schindler, and Stefan Roth, "Discrete-continuous optimization for multi-target tracking," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012, pp. 1926–1933.

[6] Mohib Ullah, Faouzi Alaya Cheikh, and Ali Shariq Imran, "Hog based real-time multi-target tracking in bayesian framework," in *13th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS) workshop, 2016*, 2016, pp. 416–422.

[7] Huilan Jiang, Jianhua Li, Dong Wang, and Huchuan Lu, "Multi-feature tracking via adaptive weights," *Neurocomputing*, vol. 207, pp. 189–201, 2016.

[8] Michael Calonder, Vincent Lepetit, Christoph Strecha, and Pascal Fua, "Brief: Binary robust independent elementary features," in *European conference on computer vision*. Springer, 2010, pp. 778–792.

[9] David G Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.

[10] Navneet Dalal and Bill Triggs, "Histograms of oriented gradients for human detection," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005, vol. 1, pp. 886–893.

[11] Zhenhua Guo, Lei Zhang, and David Zhang, "A completed modeling of local binary pattern operator for texture classification," *IEEE Transactions on Image Processing*, vol. 19, no. 6, pp. 1657–1663, 2010.

[12] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[13] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich, "Going deeper with convolutions," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 1–9.

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," *arXiv preprint arXiv:1512.03385*, 2015.

[15] "ImageNet Dataset," http://image-net.org//.

[16] Stéphane G Mallat and Zhifeng Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Transactions on signal processing*, vol. 41, no. 12, pp. 3397–3415, 1993.

[17] Matthew D Zeiler and Rob Fergus, "Visualizing and understanding convolutional networks," in *European Conference on Computer Vision (ECCV)*. Springer, 2014, pp. 818–833.

[18] Greg Welch and Gary Bishop, "An Introduction to the Kalman Filter," *In Practice*, vol. 7, no. 1, pp. 1–16, 2006.

[19] Michael Jünger, Thomas M Liebling, Denis Naddef, George L Nemhauser, William R Pulleyblank, Gerhard Reinelt, Giovanni Rinaldi, and Laurence A Wolsey, *50 years of integer programming 1958-2008: From the early years to the state-of-the-art*, Springer Science & Business Media, 2009.

[20] B Boser Le Cun, John S Denker, D Henderson, Richard E Howard, W Hubbard, and Lawrence D Jackel, "Handwritten digit recognition with a back-propagation network," in *Advances in neural information processing systems*. Citeseer, 1990.

[21] Ali Sharif Razavian, Hossein Azizpour, Josephine Sullivan, and Stefan Carlsson, "Cnn features off-the-shelf: an astounding baseline for recognition," in *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2014, pp. 806–813.

[22] John Wright, Allen Y Yang, Arvind Ganesh, S Shankar Sastry, and Yi Ma, "Robust face recognition via sparse representation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 31, no. 2, pp. 210–227, 2009.

[23] Michael Elad and Michal Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Transactions on Image processing*, vol. 15, no. 12, pp. 3736–3745, 2006.

[24] Michael Elad, "Sparse and redundant representation modelingwhat next?," *IEEE Signal Processing Letters*, vol. 19, no. 12, pp. 922–928, 2012.

[25] Geoff Davis, Stephane Mallat, and Marco Avellaneda, "Adaptive greedy approximations," *Constructive approximation*, vol. 13, no. 1, pp. 57–98, 1997.

[26] Yagyensh Chandra Pati, Ramin Rezaiifar, and PS Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," in *IEEE Conference on Signals, Systems and Computers*, 1993, pp. 40–44.

[27] Ron Rubinstein, Michael Zibulevsky, and Michael Elad, "Efficient implementation of the k-svd algorithm using batch orthogonal matching pursuit," *Cs Technion*, vol. 40, no. 8, pp. 1–15, 2008.

[28] Keni Bernardin and Rainer Stiefelhagen, "Evaluating multiple object tracking performance: The CLEAR MOT metrics," *Eurasip Journal on Image and Video Processing*, 2008.