Boosted Kernelized Correlation Filters for Event-based Face Detection

Bharath Ramesh* and Hong Yang
The N.1 Institute for Health, National University of Singapore
Email: bharath.ramesh03@u.nus.edu

Abstract

Recently, deep learning has revolutionized the computer vision field and has resulted in steep advances in the performance of vision systems for human detection and classification on large datasets. Nevertheless, these systems rely on static cameras that do not yield practical results, especially for prolonged monitoring periods and when multiple object activities occur simultaneously. We propose that event cameras naturally solve these issues at the hardware level via asynchronous, pixel-level brightness sensing at microsecond time-scale. In particular, event cameras do not output data during no-activity periods and thus data rate is drastically lowered without any additional processing. Secondly, event cameras produce disjoint spatial outputs for multiple objects without requiring segmentation or explicit background modeling. Leveraging these attractive properties, this paper presents an event-based feature learning method using kernelized correlation filters (KCF) within a boosting framework. A key contribution is the reformulation of KCFs to learn the face representation instead of relying on handcrafted feature descriptors as done in previous works. We report a high detection performance on data collected using an event camera and showcase its potential for surveillance applications. For fostering further research, we release the face dataset used in our work to the wider community[1].

1. Introduction

Over the past decade, face recognition using deep learning methods [10, 28] has matured considerably. Since detection of frontal face images under controlled settings has become easy to achieve [21], a number of recent studies have emphasized the importance of robustness to adversarial samples that compromise the system security [26, 8]. In addition, training deep learning methods for face recognition and analysis is extremely complex and time-consuming [27] and thus recent works also accommodate traditional feature extraction methods [7]. In this vein, the development of lightweight face recognition methods is also a hot topic of research [4], as CNNs that rely on network depth and thereby on powerful GPUs do not translate well to practical systems with power constraints.

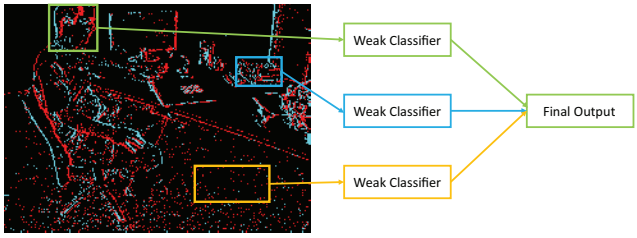

Figure 1. The proposed framework for face detection using an event camera which is viewing multiple objects, including a human (recording from [14]). Face locations and randomly sampled surrounding negative candidate locations are used to obtain weak classifiers via kernelized correlation filter outputs, which are combined using boosting to determine whether a region is a face.

A core premise of the above methods is a camera that captures intensity images at a fixed interval regardless of scene dynamics. For example, a standard camera is programmed to capture an image at 30 frames-per-second, which is a reasonable frame rate for simple standalone applications, and consequently, further analysis using multiple deep learning models for face recognition has limited application considering the energy requirements [20]. Furthermore, training and deploying deep learning models has been largely demonstrated by researchers with access to extensive computational resources. Thus, there is a parallel need to develop frugal learning frameworks aided by data-efficient sensors [22].

An alternative to the standard frame cameras, known as event cameras or silicon retinas, has been recently explored for various vision tasks [3, 6, 17, 19, 18]. Event cameras react to brightness changes sensed at an individual pixel location (x, y), which are further characterized by a timestamp t and polarity p. The polarity simply specifies the brightness change direction as a binary ON or OFF value. Additionally, these events are asynchronously transmitted with minimal latency in the order of 1-10 μs. Therefore, these

[1] DAVIS Face Dataset: https://tinyurl.com/rbppctl

cameras are well-suited during monitoring periods with no activity (zero data throughput) and also when multiple activities occur at different spatial regions of the image (local sensing paradigm). We propose the use of event cameras for public space monitoring, which includes human activity detection and action recognition.

Figure. 1 illustrates the proposed framework using kernelized correlation filters (KCF) [9] used to extract feature outputs and then fed to a binary Adaboost framework [5]: face or background. The key contribution is the reformulation of KCF filters, typically employed as discriminative trackers [29] due to their computational efficiency in the Fourier domain ($>$ 100 fps), as feature extraction modules. Thus, computational efficiency, an important emphasis of low-latency event-based vision, is also taken into account in the proposed detection framework.

2. Related Work

Standard face detection. The two prevalent approaches are either based on local visual features like SIFT [13] or on sliding window methods [25]. The feature-based approaches usually follow a three stage pipeline consisting of keypoint detection, feature extraction and model fitting (planar assumption [13, 12] or a comprehensive 3-D model [24]). Based on extensive research that has enabled detection of a wide array of objects [16, 15], these methods have also reached a high level of maturity under real-time operating needs, even on low-power devices [23]. Nonetheless, the principal drawback is the requirement to detect thousands of image features while knowing the geometrical priors of the object. This gap is neatly addressed using KCFs to learn the face geometry without explicit handcrafting.

On the other hand, sliding window-based approaches [25] search the entire image sequentially using various window sizes to decide whether it contains the object of interest. For instance, there are approximately 50,000 windows in a 320x240 resolution image. Therefore, the cascaded architecture [25] is employed for real-time performance, which reduces the number of patches to be evaluated at the final stage with full precision. However, a global sliding window approach is generally not required for event-based detection since it provides much sparser image output based on object activity. In other words, the KCF filters need to be evaluated only at locations of considerable event activity.

Event-based face detection. Being a comparatively new research area, limited works in the event-based vision domain address the problem of face detection or recognition in general. One such work is [11], which presents an intuitive approach to face detection by relying on the high-temporal resolution of the events to detect eye blinks. While the method presented in [11] is applicable to human faces in the presence of clearly discernible eye blinks, we develop a general purpose face detection method based on appearance, similar in spirit to the work on generic object detection for event cameras [17].

3. Event-based Face Detection

We exploit the seminal KCF work by Henriques *et al*. [9] that provides a theoretical framework to study generic classifiers trained online with all sub-windows (of fixed size) of a given target image, which they call "dense sampling". Before which, almost all of the proposed methods employed a sparse sampling strategy for creating real-time object representations for tracking. Obviously, the sparse sampling strategy has a reduced computation load compared to a simple dense sampling strategy. Fortunately, dense sampling of an image results in a circulant structure that permits the use of the Fast Fourier Transform (FFT) to incorporate information from all sub-windows in a single-shot manner. In addition, the KCF object representation is competitive with the state-of-the-art and runs at high frame-rates with a few lines of code.

Considering the fast and simple implementation of KCF, we apply it to event-based face representation with the following assumptions. A KCF filter outputs a higher response for the face representation compared to non-face data. Naturally, the output response of a single KCF face classifier cannot generalize to many people's faces. Therefore, we propose AdaBoost to optimize selection of weak KCF filters as long as it has an accuracy better than a random guess (0.5 in this case for face vs. non-face data).

3.1. Correlation Filters for Feature Extraction

Since event cameras do not output intensity frames, we use an event count matrix that increments the pixel count by one when it receives an event, as done in [18]. Let us consider a d-dimensional event count feature representation of a face in the training set, i.e., f be a $M \times N$ feature patch of the target. The individual face features of f are denoted by f^l where $l \in \{1, \cdots, d\}$. In order to obtain the corresponding correlation filter h of the same dimension, the following cost function can be minimized:

$$\epsilon = \| \sum_{l=1}^{d} h_l \star f_l - g \|^2 + \lambda \sum_{l=1}^{d} \| h^l \|^2 \qquad (1)$$

where g is the desired correlation output associated with the training example f. Note that the functions f, g and h are all of size $M \times N$. The desired correlation output g is constructed as a Gaussian function with its peak located at the face center in f_l. The star $\star$ denotes circular correlation. The parameter $\lambda \geq 0$ controls the impact of the regularization term.

Solving the above problem is straightforward, although it involves the optimization of a real valued function of a complex variable. After some mathematical manipulation,

a closed form expression for the KCF filter can be derived as follows:

$$H^l = \frac{\bar{G}F^l}{\sum_{k=1}^{d} \bar{F}^k F^k + \lambda} \tag{2}$$

in which the capital case letters denote the discrete Fourier transforms (DFTs) of the corresponding functions. As mentioned in [1], the regularization parameter alleviates the problem of zero-frequency components in the spectrum of f, which would lead to division by zero.

A useful face feature can be obtained by minimizing the output error over a batch of training patches. We use a batch size of twenty count matrices of one object to train a single KCF learner. However, this requires solving a $d \times d$ linear system of equations per pixel. To obtain a robust approximation for online learning applications, the numerator A_t^l and denominator B_t of the correlation filter H_t^l in (2) are updated separately:

$$\begin{aligned} A_t^l &= (1-\eta)A_{t-1}^l + \eta \bar{G}_t F_t^l \\ B_t &= (1-\eta)B_{t-1} + \eta \textstyle\sum_{k=1}^{d} \bar{F}_t^k F_t^k \end{aligned} \tag{3}$$

where η is the update rate parameter. This means each "frame"' has a decreasing effect of the latest frame to the oldest frame on the learner due to the use of the exponential moving average strategy. The correlation scores y at a rectangular region z of a feature map can be computed as:

$$y = \mathcal{F}^{-1} \frac{\sum_{l=1}^{d} \bar{A}^l Z^l}{B + \lambda} \tag{4}$$

In other words, given any feature patch z, resized to $M \times N$ if necessary, the correlation scores y are computed as $y = \mathcal{F}^{-1}\{\bar{H}_t \odot Z\}$, using the inverse DFT operator. A new target location can be estimated to be at the maximum correlation score of y, which is also the detection output location. In particular, two key factors are needed for calculating the kernel matrix k and the response with k of each KCF learner: (1) Z is the old input feature to correlate with the new test feature; (2) α is the weight matrix corresponding to the input feature.

$$\begin{aligned} \kappa(x, z) &= \langle \varphi(x), \varphi(z) \rangle \\ \hat{y} &= F^{-1}\left(F\left(\bar{k}\right) \odot F\left(\alpha\right)\right) \end{aligned} \tag{5}$$

where sizes of Z and α of different KCF learners are bound to be different depending on the size of different faces. A resize operation is done for the test feature map corresponding to each KCF filter size.

3.2. Event-based Adaboost

AdaBoost is an ensemble algorithm that outputs a weighted sum of several weak learners as a strong classifier. This boosting algorithm does not require any prior knowledge about the performance of the weak learning algorithm,

Algorithm 1 Event-based Face Detection via Adaboost

Input: KCF filters $(x_1, y_1), \cdots, (x_m, y_m)$ where $x_i \in \mathcal{X}$, $y_i \in \mathcal{Y} = \{-1, +1\}$

Initialize $D_1(i) = 1/m$

1: **for** $t = 1 : T$: **do**
2: Train weak learner using D_t
3: Obtain weak hypothesis $h_t : \mathcal{X} \to \{-1, +1\}$ with error $e_t = \Pr_{i \sim D_t}[h_t(x_i) \neq y_i]$.
4: Choose $\alpha_t = \frac{1}{2} \ln \frac{1-e_t}{e_t}$ the best weak learner
5: Update the distribution:
$$D_{t+1} = \frac{D_t(i)}{Z_t} \times \exp\left(-\alpha_t y_i h_t(x_i)\right)$$
where Z_t simply a normalization factor for D_{t+1}
6: **end for**

Output: Final hypothesis $H(x) = \text{sign}\left(\sum_{t=1}^{T} \alpha_t h_t(x)\right)$

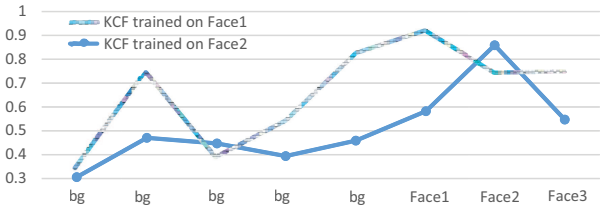

Figure 2. Average KCF responses of a desirable (Face2) vs. undesirable (Face1) weak classifier to various person's data and non-face background (BG) data.

but it has two premises. One is that the performance of each weak learner must be at least better than random guessing. The other is that the training dataset is big enough.

Algorithm 1 shows the boosting framework applied to the KCF face classifiers denoted as x_i. Firstly, initialize KCF weights evenly. Then, apply all weak learners among the training set and calculate error. Next, choose the learner with lowest error and calculate its weight. Then, update data weights and go to the next iteration until the classification error is less than a threshold or the number of iterations reaches a set point. Finally, combine all KCF weak learners into a strong classifier that can determine whether a candidate region belongs to a face and for further detection.

Fig. 2 shows sample responses from two different weak learners or KCF classifiers on various data, including non-face background (BG) data. A good candidate for the Adaboost framework is one trained on Face2, which has high responses for faces, whereas Face1 KCF produces an undesirable high response for the BG data, which is some printed signs on a cardboard box. Thus, some of the assumptions we made earlier about KCF classifier response is true only for some weak learners and the Adaboost framework is required to pick the best set of weak learners.

The subsequent problem of choosing a response thresh-

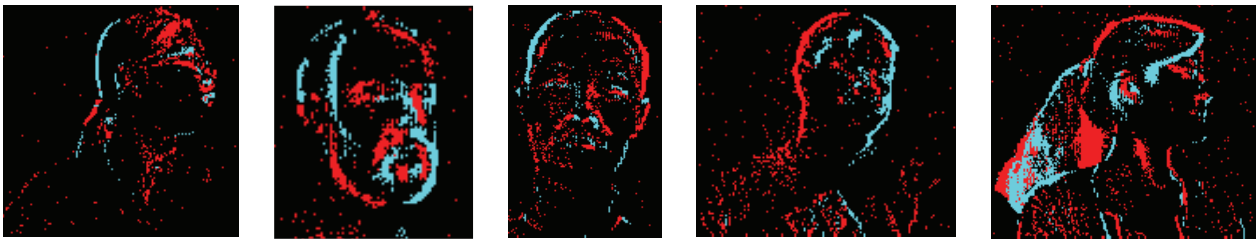

Figure 3. Samples from the DAVIS face dataset (Download at https://tinyurl.com/rbppctl).

old to robustly identify the positive class (face) is tackled by sweeping the real-valued KCF output as part of the Adaboost process exhaustively and recorded during the training process for each KCF learner.

4. Experiments

We report the proof-of-concept study used for evaluating the event-based face detection using KCF and Adaboost, as explained in the earlier section. First, data was collected using the DAVIS event camera [2], consisting of several people as shown in Fig. 3. A stationary DAVIS camera records the head motion of a person in front of it for about a minute. The subjects were asked to move their heads from frontal to side profile and also at various distances to the camera. The ground truth face positions were annotated manually for the purpose of training and testing. For the purpose of training the Adaboost classifier, non-face data was used to provide negative samples, which are usually surrounding objects as shown in Fig. 1.

The dataset was randomly divided into 70% for training and 30% for testing. Accordingly, Fig. 4 shows the variation in performance of the Adaboost face detection framework as weak learners are added (Alg. 1). The detection accuracy increases as expected and tapers off after accumulating about 50 KCF weak learners from different people. Note that with just one optimal weak learner selected by Adaboost, the system performance is already about 70%. This showcases the ability of the proposed event-based framework to distinguish between faces and non-faces.

Further experiments using a lower train-test split ratio of 10-90% still resulted in a detection performance of 84.42%, which shows the capacity of the KCF weak learners selected by Adaboost to generalize other faces while rejecting surrounding background information. Subsequently, we pitted the faces against each other (instead of BG) and the recognition accuracy was 35%, which implies that the KCFs do not learn fine-grained features to distinguish between the faces.

Finally, we report comparison with the traditional Adaboost framework for face detection as it is a closely related method [25]. The standard Viola-Jones detector tested on the DAVIS face dataset (70-30% split) obtains 40.4% detection accuracy, which is much lower compared to our results. This is probably because of the low resolution of the event

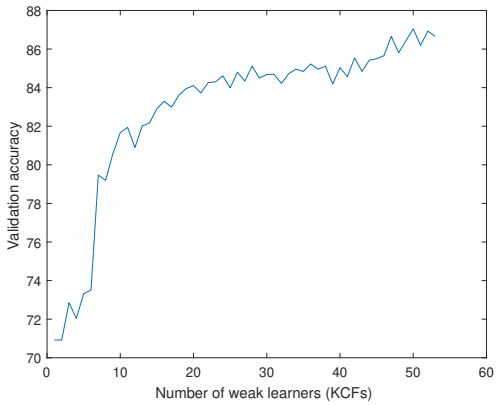

Figure 4. Impact of weak learners on the validation accuracy.

camera as well as the lack of RGB information.

Due to the low resolution of the DAVIS camera (240×180), it remains harder to test wide field-of-view (FOV) recordings, although the DAVIS face dataset is suitable for developing other face recognition systems. Newer event cameras developed by CelePixel with 1280×800 image resolution can address the FOV problem easily. Besides, the study presented here is preliminary, since only the classification of candidate regions-of-interest has been presented while the method to obtain them has been assumed. This will be one of the future works in addition to mining negative samples for the Adaboost framework in the general case of recordings with multiple faces in the field-of-view.

5. Conclusion

This paper presented one of the first event-based face detection frameworks by using kernelized correlation filters within an Adaboost classification scheme. The main contribution was the reformulation of KCFs to learn the face representation instead of relying on handcrafted feature descriptors as done in earlier works. We reported a high detection performance (87%) on the DAVIS face dataset, even at low train-test split ratios (84%). Finally, we reported 2x higher performance compared to the standard Viola-Jones face detector. For fostering further research, our dataset has been made publicly available.

References

[1] D. S. Bolme, J. R. Beveridge, B. Draper, Y. M. Lui, et al. Visual object tracking using adaptive correlation filters. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 2544–2550. IEEE, 2010. 3

[2] C. Brandli, L. Muller, and T. Delbruck. Real-time, high-speed video decompression using a frame- and event-based davis sensor. In *2014 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 686–689, June 2014. 4

[3] A. Censi and D. Scaramuzza. Low-latency event-based visual odometry. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 703–710. IEEE, 2014. 1

[4] J. Deng, J. Guo, D. Zhang, Y. Deng, X. Lu, and S. Shi. Lightweight face recognition challenge. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 0–0, 2019. 1

[5] R. Duda, P. Hart, and D. Stork. Pattern classification. *New York: John Wiley, Section*, 10:l, 2001. 2

[6] G. Gallego, J. E. Lund, E. Mueggler, H. Rebecq, T. Delbruck, and D. Scaramuzza. Event-based, 6-dof camera tracking for high-speed applications. *arXiv preprint arXiv:1607.03468*, 2016. 1

[7] M.-I. Georgescu, R. T. Ionescu, and M. Popescu. Local learning with deep and handcrafted features for facial expression recognition. *IEEE Access*, 7:64827–64836, 2019. 1

[8] G. Goswami, N. Ratha, A. Agarwal, R. Singh, and M. Vatsa. Unravelling robustness of deep learning based face recognition against adversarial attacks. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018. 1

[9] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista. High-speed tracking with kernelized correlation filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(3):583–596, 2015. 2

[10] G. Hu, Y. Yang, D. Yi, J. Kittler, W. Christmas, S. Z. Li, and T. Hospedales. When face recognition meets with deep learning: an evaluation of convolutional neural networks for face recognition. In *Proceedings of the IEEE international conference on computer vision workshops*, pages 142–150, 2015. 1

[11] G. Lenz, S. Ieng, and R. Benosman. Event-based dynamic face detection and tracking based on activity. *CoRR*, abs/1803.10106, 2018. 2

[12] V. Lepetit, P. Lagger, and P. Fua. Randomized trees for real-time keypoint recognition. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 775–781, 2005. 2

[13] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004. 2

[14] E. Mueggler, H. Rebecq, G. Gallego, T. Delbruck, and D. Scaramuzza. The event-camera dataset and simulator: Event-based data for pose estimation, visual odometry, and SLAM. *The International Journal of Robotics Research*, 36(2):142–149, 2017. 1

[15] S. Obdrzalek and J. Matas. Sub-linear indexing for large scale object recognition. In *BMVC*, pages 1–10, 2005. 2

[16] J. Pilet and H. Saito. Virtually augmenting hundreds of real pictures: An approach based on learning, retrieval, and tracking. In *Virtual Reality Conference (VR), 2010 IEEE*, pages 71–78. IEEE, 2010. 2

[17] B. Ramesh, H. Yang, G. M. Orchard, N. A. Le Thi, S. Zhang, and C. Xiang. DART: Distribution Aware Retinal Transform for Event-based Cameras. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2019. 1, 2

[18] B. Ramesh, S. Zhang, Z. W. Lee, Z. Gao, G. Orchard, and C. Xiang. Long-term object tracking with a moving event camera. In *BMVC*, 2018. 1, 2

[19] H. Rebecq, T. Horstschafer, G. Gallego, and D. Scaramuzza. Evo: A geometric approach to event-based 6-dof parallel tracking and mapping in real-time. *IEEE Robotics and Automation Letters*, 2016. 1

[20] M. Shafique, T. Theocharides, C.-S. Bouganis, M. A. Hanif, F. Khalid, R. Hafız, and S. Rehman. An overview of next-generation architectures for machine learning: Roadmap, opportunities and challenges in the iot era. In *2018 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, pages 827–832. IEEE, 2018. 1

[21] X. Sun, P. Wu, and S. C. Hoi. Face detection using deep learning: An improved faster rcnn approach. *Neurocomputing*, 299:42–50, 2018. 1

[22] K. Sundararajan and D. L. Woodard. Deep learning for biometrics: A survey. *ACM Comput. Surv.*, 51(3):65:1–65:34, May 2018. 1

[23] S. Taylor and T. Drummond. Multiple target localisation at over 100 fps. In *BMVC*, pages 1–11, 2009. 2

[24] L. Vacchetti, V. Lepetit, and P. Fua. Stable real-time 3d tracking using online and offline information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(10):1385–1391, 2004. 2

[25] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages I–511, 2001. 2, 4

[26] H. Wang, Y. Wang, Z. Zhou, X. Ji, D. Gong, J. Zhou, Z. Li, and W. Liu. Cosface: Large margin cosine loss for deep face recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5265–5274, 2018. 1

[27] Q. Wang and G. Guo. Benchmarking deep learning techniques for face recognition. *Journal of Visual Communication and Image Representation*, 65:102663, 2019. 1

[28] Y. Wen, K. Zhang, Z. Li, and Y. Qiao. A discriminative feature learning approach for deep face recognition. In *European conference on computer vision*, pages 499–515. Springer, 2016. 1

[29] K. Zhang, L. Zhang, Q. Liu, D. Zhang, and M.-H. Yang. Fast visual tracking via dense spatio-temporal context learning. In *Proceedings of European Conference on Computer Vision*, pages 127–141. Springer, 2014. 2