JOINT OPTIMIZATION OF BACKGROUND SUBTRACTION AND OBJECT DETECTION FOR NIGHT SURVEILLANCE

Congcong Li, Chih-Wei Lin*†, Shiaw-Shian Yu*†, Tsuhan Chen*

* Cornell University, USA † Industrial Technology Research Institute, Taiwan

ABSTRACT

Detecting foreground objects for night surveillance videos remains a challenging problem in scene understanding. Though many efforts have been made for robust background subtraction and robust object detection respectively, the complex illumination condition in night scenes makes it hard to solve each of these tasks individually. In practice, we see these two tasks are coupled and can be combined to help each other. In this work, we apply a recently proposed algorithm – Feedback Enabled Cascaded Classification Models (FECCM) – to combine the background subtraction task and the object detection task into a generic framework. The proposed framework treats each classifier for the respective task as a 'blackbox', thus allows the usage of most existing algorithms as one of the classifiers. Experiment results show that the proposed method outperforms a state-of-the-art background subtraction method and a state-of-the-art object detection method.

Index Terms— Optimization, object detection, background subtraction, surveillance

1. INTRODUCTION

Detecting foreground objects is an important step for analyzing night surveillance videos. Though many efforts have been made for developing robust background subtraction algorithms and robust object detection algorithms, both the background subtraction task and object detection task remain difficult for night surveillance. Under night scenes, many existing background subtraction methods and object detection methods suffer much from either heavy false alarm due to dramatic lighting changes or missing detection as the foreground color is very closed to the background in local due to low contrast, as shown in Figure 1. In this work, our goal is to achieve better performance in both background subtraction and object detection for night surveillance videos.

Background Subtraction: Background subtraction plays an important role in many applications in video surveillance area, such as key-frame extraction, video summarization, object detection, etc. Many efforts have been made to improve the performance of subtracting the unmoving background, e.g. [1, 2, 3, 4]. The traditional pixel level methods like [1, 4] model the background as a set of independent pixel processes, which lose the spatial context information and often end up with noisy detection. Therefore many methods are proposed to utilize the spatial information between pixels [2, 3], or to utilize temporal information [5] to better model the background in a scene, or combine both methods [6]. However, in a night outdoor scene, the current existing methods still suffer much from the following problems: 1) heavy false alarm due to dramatic lighting changes and reflections on other static objects; 2) missing detection due to the condition that the foreground is very similar to the background in local due to low contrast. The spatio-temporal modeling method in [6] tries to combine the spatial information and temporal information to better model the background in night outdoor scene and achieves some improvement over the previous spatial-only or temporal-only methods. This model assumes a background patch under various lighting conditions lies in a lower-dimension subspace than a patch under foreground occlusion. Therefore, the model is less sensitive to the lighting changes. However, in practical testing, we notice that this model still has problems: 1) it fails when there is strong surface reflection in the environment; 2) it is sensitive to the selection of parameters. 3) It could not handle well with the gradual change of the global illumination across the scene.

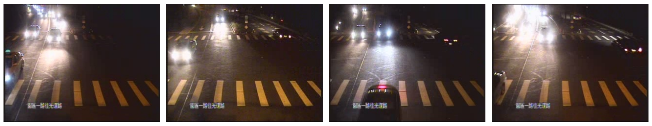

Fig. 1. Difficult examples of detecting foreground objects in night scenes due to: 1) Dramatic illumination changes; 2) low contrast between foreground and background.

Object Detection: We consider another way to detect the foreground objects by training detectors for objects of some specific categories that can appear in the surveillance videos. In this work, we focus on outdoor night surveillance videos, where objects of interest mainly include: cars, motorbikes and persons. Although the objects in the night scene look very different from those of the same category in the daytime, they still share some common visual properties while in the same type of scene. For example, the two turn-on headlights are helpful hints to recognize a car in the night scene. These observations suggest us to train some object detectors of the night scene in order to better find out the object region and ex-

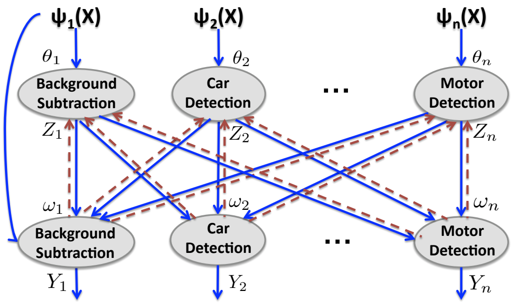

Fig. 2. Our instantiation of the FECCM model to combine background subtraction task and object detection tasks. ($\forall i \in \{1, 2, \ldots, n\}$ $\Psi_i(X)$ = Features corresponding to the specific task extracted from image X, Z_i = Output of the corresponding task in the first stage parameterized by θ_i, Y_i = Output of the corresponding task in the second stage parameterized by ω_i).

clude the lighted background. Since some parts of the object are more discriminative than others for recognizing the object, the part-based object model like [7] can be a reasonable choice for building an object detector.

Combining Tasks: The background subtraction task and the object detection task individually can help detect the foreground regions to some extent. In practice, we see that these two tasks are coupled – for example, if we know many pixels inside a region belong to the foreground, it is more likely to have an object of interest located in that region; reversely, if we detect a bounding box for an object of interest, it could serve as a strong indicator that the pixels inside the box are foreground pixels. Therefore, sharing information between these two tasks can help to improve each other.

Our contribution: We propose to combine the background subtraction task and the object detection task into one framework and leverage each of them to help the other. We apply a recently proposed model called Feedback Enabled Cascaded Classification Model (FECCM) [8] to combine these two tasks. The FECCM method jointly optimizes multiple tasks being considered, and treats each classifier for a respective task as a black-box, which allows the usage of any existing methods in the black-box. Different from [8] which focuses on multiple tasks in scene understanding, we adapt the model to the area of night surveillance videos in this work. We also show that our method outperforms a state-of-the-art background subtraction method and a state-of-the-art object detection method for night surveillance.

2. ALGORITHM

In this section we introduce our instantiation of the Feedback Enabled Cascaded Classification Model for night surveillance application. We consider the following tasks: background subtraction and detections of objects including car, person, and motorbike.

Our instantiation of the model to combine the above tasks is shown in Figure 2. The model is built in the form of a two-layer cascade. The first layer consists of an instantiation of each task with the image features as input. The second layer is a repeated instantiation of each of the tasks with the first layer task outputs as well as the image features as inputs. In the following we briefly review the learning and inference algorithms in [8]. Please refer to [8] for more details.

We want to model the conditional joint log likelihood of all the task outputs, i.e., $\log P(\mathcal{Y}|X)$, where $\mathcal{Y}$ indicates the set $Y_1, Y_2, \ldots, Y_n$, and X is an image in a training set Γ.

$$\log \prod_{X \in \Gamma} P(\mathcal{Y}|X; \Theta, \Omega) \tag{1}$$

where $\Theta = \{\theta_1, \ldots, \theta_n\}$, $\Omega = \{\omega_1, \ldots, \omega_n\}$ are internal parameters of the black-box classifiers used in the model.

To incorporate the first layer outputs, Equation 1 can be extended as follows.

$$= \sum_{X \in \Gamma} \log \sum_{\mathcal{Z}} P(Y_1, \ldots, Y_n, \mathcal{Z}|X; \Theta, \Omega) \tag{2}$$

$$= \sum_{X \in \Gamma} \log \sum_{\mathcal{Z}} \prod_{i=1}^{n} P(Y_i|\Psi_i(X), \mathcal{Z}; \omega_i) P(Z_i|\Psi_i(X); \theta_i) \tag{3}$$

where $\mathcal{Z}$ indicates the set $Z_1, Z_2, \ldots, Z_n$.

However, the summation inside the log makes it difficult to learn the parameters. Heitz et al [9] gives an approximate solution by optimizing classifiers on each layer independently (without considerations for other layers). This has a drawback that there is no feeding back information from later classifiers to earlier classiers during training. Motivated by the Expectation Maximization [10] algorithm, the FECCM method uses an iterative algorithm where we first fix the latent variables Z_i's and learn the parameters in the first step (Feed-forward step), and estimate the latent variables Z_i's in the second step (Feed-back step). Z_i's are initialized to the ground truth of the respective tasks. We then iterate between these two steps.

Feed-forward Step: In this step, we assume that the latent variables Z_i's are known (and Y_i's are known anyway because they are the ground-truth). Then optimizing Equation 3 over the parameters can be nicely broken down into the following sub-problems of training the individual classifier for the respective tasks:

$$\max_{\omega_i} \sum_{X \in \Gamma} \log P(Y_i|\Psi_i(X), Z_1, \ldots, Z_n; \omega_i) \tag{4}$$

$$\max_{\theta_i} \sum_{X \in \Gamma} \log P(Z_i|\Psi_i(X); \theta_i) \tag{5}$$

Note that we can use the same training algorithm as the original black-box classifier to solve these sub-problems.

Feed-back Step: In this step, we estimate the values of the latent variables Z_i's assuming that the parameters are fixed (and Y_i's are given because the ground-truth is available). We perform MAP inference on Z_i's. Using Equation 3, we get the following optimization problem for the feed-back step:

$$\max_{Z_1, \ldots, Z_n} \log P(Y_1, \ldots, Y_n, Z_1, \ldots, Z_n | X; \theta_1, \ldots, \theta_n, \omega_1, \ldots, \omega_n)$$

$$\Leftrightarrow \max_{Z_1, \ldots, Z_n} \sum_{i=1}^{n} \log P(Z_i|\Psi_i(X); \theta_i) + \log P(Y_i|\Psi_i(X), \mathcal{Z}; \omega_i) \tag{6}$$

This maximization problem requires that we have access to the characterization of the individual black-box classifiers in a probabilistic form. We do this by taking the output of the classifiers and modeling their log-odds as a Gaussian (partly motivated by variational approximation methods). Parameters of the Gaussians are empirically estimated when the actual probabilistic form is not available.

In some cases, the classifier log-likelihoods in the problem in Equation 6 actually turn out to be convex. For example, if the individual classifiers are linear or logistic classifiers, the minimization problem is convex and can be solved by using a gradient descent (or any similar method).

Inference: Similar to the learning process, the inference is conducted in a feed-forward manner: solve the first layer outputs and then solve the second layer outputs. Given the structure of our directed graph, the outputs for different classifiers on the same layer are independent given their inputs and parameters. Therefore, we have

$$\hat{Z}_i = \underset{Z_i}{\operatorname{argmax}} \log P(Z_i|\Psi_i(X), \theta_i), i = 1, \dots, n \quad (7)$$

$$\hat{Y}_i = \underset{Y_i}{\operatorname{argmax}} \log P(Y_i|\hat{\mathcal{Z}}, \Psi_i(X), \omega_i), i = 1, \dots, n \quad (8)$$

This approximate inference allows us to use the internal inference function of the black-box classifiers without knowing its inner workings. It is tractable since its complexity is no more than constant times the complexity of inference in the original classiers.

3. IMPLEMENTATION

In this section we will introduce the classifiers used in the framework shown in Figure 2 and the state-of-the-art method [6] used for comparison.

Background Subtraction. For the first-layer background subtraction, we build a gaussian model for the gray-scale value of each pixel when being background. For a new frame, it is considered to be a foreground pixel when its difference to the model mean is larger than a threshold T (refer to codes for the exact value). The model is updated dynamically as follows.

$$\theta^l(p) = (1 - \alpha) \times \theta^{l-1}(p) + \alpha \times I^l(p) \text{ if } p \in \text{foreground} \quad (9)$$

$$\theta^l(p) = (1 - \beta) \times \theta^{l-1}(p) + \beta \times I^l(p) \text{ if } p \in \text{background} \quad (10)$$

where p indicates a pixel in the image, $\theta^l(p)$ is the Gaussian mean of the background model for the pixel p at the l^{th} frame, $\theta^{l-1}(p)$ is the Gaussian mean of the background model for the pixel p at the $(l-1)^{th}$ frame, and $I^l(p)$ is the gray-scale value of the pixel p in the l^{th} frame. α and β are the update factors, where β is set to be larger than α (In our implementation, $\alpha = 0.05$ and $\beta = 0.2$).

For the second-layer background subtraction, we use a logistic classifier, which classifies a pixel as foreground or background. The input feature vector includes the original feature input of the first-layer background subtraction (the absolute difference between the pixel value and the Gaussian mean), the binary output of the first-layer background subtraction, and the binary output at the pixel from each of the first-layer object detectors.

Object Detection. For the first-layer object detectors, we use histogram of oriented gradients (HOG) features [11] and apply the deformable-parts-based model in [7]. The deformable-parts-based model contains a mixture of components, allowing for better modeling of the variety of objects within a category. Each component contains a coarse root-filter that serves as a global template for the object, and higher resolution part-filters for different localized parts of the object. In our implementation, we use 8 part-filters. The spatial locations of the object and parts is modeled via a star-graph. The deformable-parts-model is trained discriminatively via a latent SVM. A detailed description of the model can be found in [7]. In our implementation, we first divide the training images into 2 groups based on the time period. For each group, we further divide the object boxes into 2 sub-groups based on the aspect ratio of the ground-truth bounding boxes. For each of these sub-groups, we generate a left-right flipped version of each image and then use them to train 2 components respectively for the left and right poses of the object. Therefore, we have 8 components in total for an object template.

On the second layer, an object detector is a classifier which re-scores all the candidate boxes detected from the first-layer object detector with an extremely low threshold. The classifier is a RBF-kernel SVM classifier, whose input includes the top-left and bottom-right coordinates $(x1, y1, x2, y2)$ of a candidate box, the first-layer object detector output score for the candidate box, and a score which is the mean value of the first-layer background subtraction outputs of all pixels inside the candidate box.

Subspace Background Subtraction Based on Spatio-temporal Patches. This is our implementation of the method proposed in [6]. We add two more steps to improve the original algorithm: (1) For patches that are decided to be foreground by the original algorithm, we look at their standard derivation over the past 25 frames. If there is no much change in the past 25 frames, We decide this patch belongs to the background and will be used to update the background model. This helps to eliminate the ghost effect. When some objects which are part of the background in previous frame start to move, dramatic changes would happen in some patches and a new background model is needed to be built for those patches. However, according to the original algorithm in [6], those patches with dramatic changes would not be used to update the model. (2) We remove connected regions whose area is too small, in order to remove some noisy detection.

4. EXPERIMENTS

Dataset: We use the road surveillance dataset and the gate entrance dataset built by Industrial Technology Research In-

Table 1. Performance of background subtraction

	F1 measure
Pixel-based Gaussian Model	0.385
Subspace Method [6]	0.514
Our Method	**0.622**

Table 2. Performance of object detection

	AP (Average Precision)		
	Car	Person	Motorbike
Part-based Method [7]	0.556	0.310	0.223
Our Method	**0.610**	**0.352**	**0.425**

stitute for experiments. For each dataset, We use 5 videos for training and 10 videos for testing (3000 – 6000 frames per video) . For each frame, the foreground regions are labeled with bounding boxes and object categories.

Evaluation: We evaluate the object detection performance via the average precision (AP) of precision-recall curves as in [12]. We evaluate the final foreground detection output with the F1 measure, computed as follows.

$$precision = \frac{1}{N}\sum_i \frac{GT^i \cap Y_1^i}{\sum_p Y_1^i(p)} \tag{11}$$

$$recall = \frac{1}{N}\sum_i \frac{GT^i \cap Y_1^i}{\sum_p GT^i(p)} \tag{12}$$

$$F1 = \frac{2 \times precision \times recall}{precision + recall} \tag{13}$$

where GT^i is the ground-truth foreground map for the i^{th} test image, and Y_1^i is the detected foreground map for the i^{th} test image. $GT^i(m,n)$ equals to 1 when the pixel p belongs to foreground, otherwise equals to 0.

Table 1 and Table 2 give results for background subtraction and object detection respectively. Note that with one single model, our method outperforms a state-of-the-art background subtraction method in [6] and a state-of-the-art object detection method in [7]. Some visualized results from the proposed algorithm are given in Figure 3.

5. CONCLUSION

In this work, we use a generic model – Feedback Enabled Cascaded Classification Model – to combine background subtraction and object detection for improved foreground detection in night surveillance. The proposed method treats each classifier for the respective task as a 'black-box', thus allows using any existing algorithm as one of the classifiers in the model. Experiments on real data show that our method outperforms a state-of-the-art background subtraction method specifically designed for night scenes and a popular object detection method. In the future, we would like to try more advanced algorithms in the black-box classifiers, to further improve the the final output.

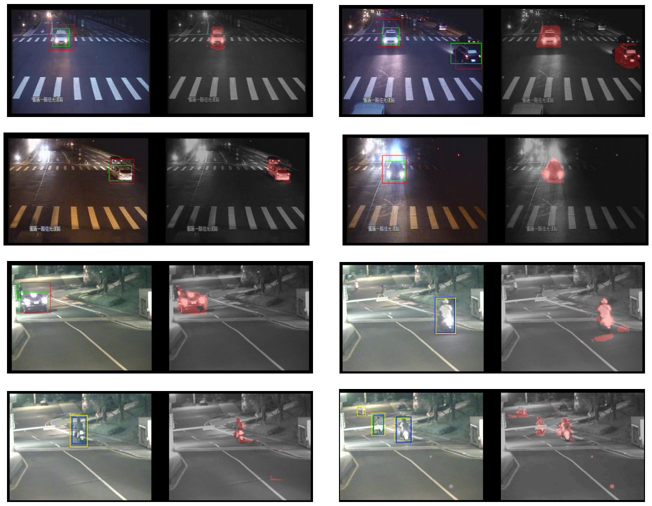

Fig. 3. Examples of the results on surveillance videos in the two ITRI datasets. Each row corresponds to 2 examples. In each example, the left image shows the groundtruth foreground objects (green for "car", blue for "motorbike") and detected objects (red for "car", yellow for "motorbike"), and the right image shows the detected foreground pixels (in pink mask). Best viewed in color.

6. REFERENCES

[1] C. Stauffer and W.E.L. Grimson, "Adaptive background mixture models for real-time tracking," in *CVPR*, 1999.

[2] M. Cristani, M. Bicego, and V. Murino, "Integrated region and pixel-based approach to background modeling," in *MOTION*, 2002.

[3] M. Heikkil and M. Pietik, "A texture-based method for modeling the background and detecting moving objects," *IEEE Trans. PAMI*, vol. 28, no. 4, pp. 657–661, April 2006.

[4] W. Z. Hu, H. F. Gong, S. C. Zhu, and Y. T. Wang, "An integrated background model for video surveillance based on primal sketch and 3d scene geometry," in *CVPR*, 2008.

[5] L. Wixson, "Detecting salient motion by accumulating directionally consistent flow," *IEEE Trans. PAMI*, vol. 22, no. 8, pp. 774–780, August 2000.

[6] Y. Zhao, H. Gong, L. Lin, and Y. Jia, "Spatial-temporal patches for night background modeling by subspace learning," in *ICPR*, 2008.

[7] P. Felzenszwalb, D. McAllester, and D. Ramanan, "A discriminatively trained, multiscale, deformable part model," in *CVPR*, 2008, pp. 1–8.

[8] C. Li, A. Kowdle, A. Saxena, and T. Chen, "Towards holistic scene understanding: Feedback enabled cascaded classification models," in *NIPS*, 2010.

[9] G. Heitz, S. Gould, A. Saxena, and D. Koller, "Cascaded classification models: Combining models for holistic scene understanding," in *NIPS*, 2008.

[10] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *J of Royal Stat. Soc., Series B*, vol. 39, no. 1, pp. 1–38, 1977.

[11] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *CVPR*, 2005.

[12] M. Everingham, L.V. Gool, C. Williams, J. Winn, and A. Zisserman, "The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results," http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html.