STATISTICAL BACKGROUND SUBTRACTION BASED ON IMBALANCED LEARNING

*Xiang Zhang** *Zhi Liu*† *Hongsheng Li** *Xu Zhao*‡ *Ping Zhang**

*School of Electronic Engineering, University of Electronic Science and Technology, China
†School of Communication and Information Engineering, Shanghai University, China
‡School of Electronic Information, Shanghai Jiaotong University, China
uestchero@uestc.edu.cn, liuzhisjtu@163.com, lihongsheng@gmail.com,
zhaoxu@sjtu.edu.cn, pingzh@uestc.edu.cn

ABSTRACT

In this paper, we study the class imbalance problem in statistical background subtraction. Firstly, we discuss the imbalance essence in background subtraction, and conclude that foreground and background are inherently imbalanced. Secondly, following the imbalanced learning strategy in machine learning, we present a spatio-temporal over-sampling method to resolve the class imbalance in background subtraction. Our method densely generate synthesized foreground samples in compact 3D spatio-temporal domain. Those generated samples could reduce the imbalance level between foreground and background from both quantity and quality, and therefore contribute to improvement of detection performance. We also define a new index to measure the change of imbalance level during over-sampling. Experiments are conducted on public datasets to demonstrate the effectiveness of our method.

Index Terms— imbalanced learning, class imbalance, background subtraction, moving object detection

1. INTRODUCTION

Moving object detection, also called background subtraction, is a key step in many computer vision applications. Interferences in imaging environment are generally recognized as challenges for moving object detection. Toyama et al. [1] in an early work conclude nine such problems, e.g., light switch, waving trees, sleeping and waking persons, etc. Recently, Brutzer et al. [2] list seven challenges including gradual and sudden illumination changes, dynamic background, camouflage, shadows, bootstrapping and video noise.

This work was supported by National Natural Science Foundation of China (No. 61105001, No. 61171144 and No. 61308102), the Key (Key grant) Project of Chinese Ministry of Education (No. 212053), the Innovation Program of Shanghai Municipal Education Commission (No. 12ZZ086), the Chinese Postdoctoral Science Foundation (No. 2013M531946), the Sichuan Provincial Key Technology Research and Development Program (No. 2014GZX0009), the Fundamental Research Funds for the Central Universities of China (No. ZYGX2013J059), and a Marie Curie International Incoming Fellowship within the 7th European Community Framework Programme under Grant No. 299202.

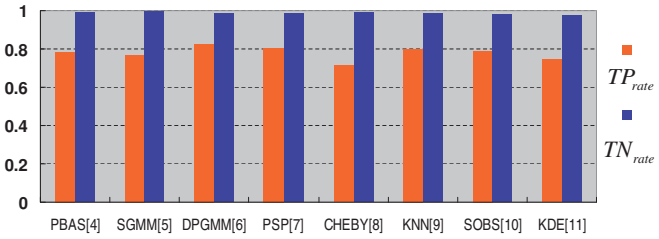

Fig. 1. TP_{rate} and TN_{rate} of eight state-of-the-art moving object detection algorithms on the CDW2012 database.

Table 1. Global imbalance level on CDW2012 database.

subset	1	2	3	4	5	6
η	0.031	0.014	0.045	0.038	0.053	0.081

We argue that the class imbalance is another key issue in background subtraction. In machine learning [3], class imbalance refers to the case that the majority (or negative) class is represented by a large number of samples while the minority (or positive) class by only a few. Classifiers trained with such skewed data tend to generate results with high true negative rate and low true positive rate. We found that class imbalance also exists in background subtraction. We define global imbalance degree in a video as $\eta = sum(F)/sum(B)$, where $sum(F)$ and $sum(B)$ are the sums of foreground and background pixels, respectively. Then we compute η on the CDW2012 database (www.changedetection.net, this database consists of six subsets), which is shown in Table 1.

Table 1 reveals that video data indeed is imbalanced, where foreground and background are the minority class and majority class, respectively. Define TP_{rate} and TN_{rate} as

$$TP_{rate} = \frac{TP}{TP+FN} \quad \text{and} \quad TN_{rate} = \frac{TN}{TN+FP},$$

where *TP*, *FP*, *TN* and *FN* are total numbers of true positives, false positives, true negatives and false negatives, respectively. The performance of eight state-of-the-art moving object detection algorithms [4]-[11] on CDW2012 database is shown in Fig. 1, where all algorithms exhibit high TN_{rate} and rela-

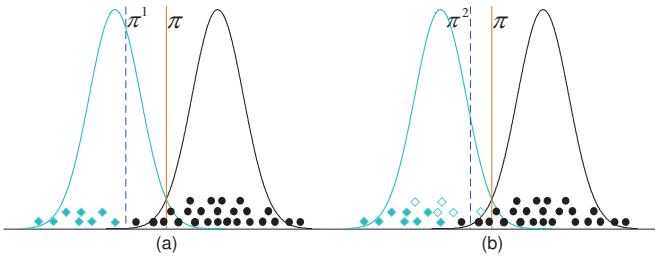

Fig. 2. (a) Illustration of the working mechanism of class imbalance in 2D space. (b) An example of synthetic over-sampling.

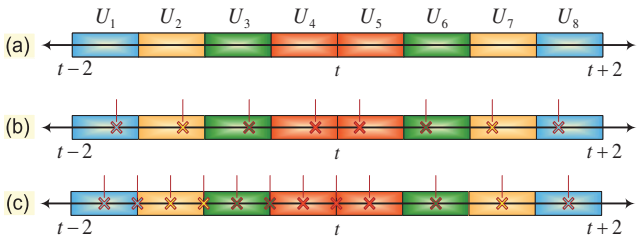

Fig. 3. (a): Division of resampling interval $(t-2, t+2]$. (b) and (c): $r(n)$ (marked with x-marks) of synthesized frames with $N=2$ and $N=3$, respectively.

tively low TP_{rate}. Through joint observation of Table 1 and Fig. 1, it is directly linked to the class imbalance problem.

The essence of class imbalance in machine learning is as follows [13]. Given an imbalanced training set $\mathbf{\Omega} = \mathbf{\Omega}^+ \cup \mathbf{\Omega}^-$, and two independent distributions they are drawn from: $\mathcal{P}$ and $\mathcal{G}$. Since the number of samples from $\mathcal{G}$ is larger than that from $\mathcal{P}$, it is likely to frequently encounter outlying negative samples during the learning process. As a result, the induced classifier tends to be skewed toward the positive class. In Fig. 2a, filled diamonds and circles represent positive and negative samples, respectively, and the corresponding latent Gaussians from which these samples are drawn are also shown. The blue dotted line (π^1) is the hypothesis induced over training samples, while the solid line (π) depicts the optimal separator of the underlying distributions. It can be clearly seen that π^1 is skewed toward the positive class. A principled way of imbalanced learning is over-sampling. An example of over-sampling is illustrated in Fig. 2b, where hollow diamonds represent synthesized positive samples. The dotted line (π^2) is the hypothesis induces over the training set after over-sampling. In this case, π^2 is more close to the optimal separator than the hypothesis π^1 before over-sampling.

The imbalance essence in background subtraction is different from the general cases. In background subtraction, positive samples $\mathbf{\Omega}^+$ and negative samples $\mathbf{\Omega}^-$ are composed of segmented foreground and background in the last τ_f and τ_b frames, respectively. Plentiful background samples generally are available by long observation of the scene, while foreground often has short exposure in the scene, resulting in relatively few foreground samples. Such difference in quantity is called relative imbalance. Moreover, foreground often exhibits much greater change than background, causing the current foreground cannot be effectively represented by those foreground samples in past frames. Such qualitative difference between foreground and background samples is known as absolute rarity [12], where the positive class is poorly characterized by low-quality positive samples.

Following the imbalanced learning strategy, we present a spatio-temporal over-sampling method for background subtraction. With past segmentations as references, our method densely generates the synthesized foreground samples to resolve relative imbalance. Those synthesized samples are compactly gathered in spatio-temporal region centered at the current instant, and hence also contribute to moderating absolute rarity. Finally, based on the rebalanced training set, nonparametric foreground and background models are learned and combined in an energy minimization framework to competitively classify new observations. We also define a new index to measure the change of imbalance level during resampling. Experiments are conducted on the public data for comparison of our method with state-of-the-art algorithms.

The rest paper is organized as follows. Section 2 presents details of the spatio-temporal over-sampling method. Modeling and classification are described in Section 3. Experiments are shown in Section 4, followed by conclusion in Section 5.

2. SPATIO-TEMPORAL OVER-SAMPLING

As shown in Fig. 2b, the missed distribution information can be partially recovered by synthetic over-sampling. Our method follows this line but works in 3D spatio-temporal domain. Spatio-temporal over-sampling is achieved via two steps. First the instants of all synthesized frames are computed (temporal over-sampling), and then new samples are generated in each synthesized frame (spatial over-sampling).

First we give some notations. Let $\mathbf{Z}^t = \{\mathbf{z}_i^t\}$ be a frame at instant t, and (x_i, y_i) the coordinate of $\mathbf{z}_i^t$. Sometimes we omit the superscript and subscript and use $\mathbf{z}(x, y)$ instead of $\mathbf{z}_i$ in case of no confusion. Let $\mathbf{F}^t$ and $\mathbf{B}^t$ be segmented foreground and background of $\mathbf{Z}^t$, respectively. We call $\mathbf{F}^t$ a foreground frame, $\mathbf{B}^t$ a background frame, and all synthesized samples at the same instant a synthesized frame.

2.1. Temporal Over-sampling

We just use foreground frames in the last $\tau_f = 4$ frames as references of over-sampling, since foreground only keep coherence in neighboring frames. All synthesized frames are limited in a compact interval $(t-2, t+2]$ to guarantee the representativeness of the generated samples.

First we evenly divide $(t-2, t+2]$ into eight subintervals: $U_k = (t + k/2 - 5/2, t + k/2 - 2], k = 1, ..., 8$. Then these subintervals are equally assigned to reference frames for resampling. Interval division is shown in Fig. 3a. We assign $U_1 \cup U_8$ to $\mathbf{F}^{t-4}$, $U_2 \cup U_7$ to $\mathbf{F}^{t-3}$, $U_3 \cup U_6$ to $\mathbf{F}^{t-2}$, $U_4 \cup U_5$ to $\mathbf{F}^{t-1}$, respectively. It can be seen that the time order of subintervals is consistent with reference frames.

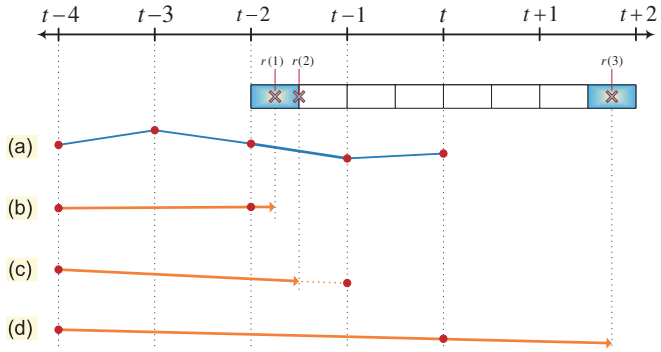

Fig. 4. Illustration of computation of motion vectors. (a): The pixel trajectory of a foreground pixel. (b) to (d): Calculation of motion vectors to instant $r(1)$, $r(2)$ and $r(3)$, respectively.

Next we compute the time instants of synthesized frames. Assuming N frames $\{\tilde{\mathbf{F}}^{r(n)}|n=1,...,N\}$ are synthesized for a particular reference frame, and they should be evenly placed in resampling subintervals. Here $r(n)$ is the time instant of $\tilde{\mathbf{F}}^{r(n)}$, and it is computed as follows.

For $\{\tilde{\mathbf{F}}^{r(n)}\}$ with $\mathbf{F}^{t-1}$ as the reference frame,

$$r(n) = t + \frac{n}{N+1} - \frac{1}{2}. \tag{1}$$

For $\{\tilde{\mathbf{F}}^{r(n)}\}$ with $\mathbf{F}^{t-s}(s \neq 1)$ as the reference frame,

$$r(n) = \begin{cases} t + \dfrac{n}{N+1} - \dfrac{s}{2} & \text{if } n \leq \dfrac{N+1}{2} \\ t + \dfrac{n}{N+1} + \dfrac{s-1}{2} & \text{otherwise.} \end{cases} \tag{2}$$

Time instants of all synthesized frames with $N=2$ and $N=3$ are shown in Fig. 3b and Fig. 3c, respectively. Taking $\mathbf{F}^{t-1}$ as an example, we can see that its resampling intervals $U_4 \cup U_5$ indeed are evenly divided by synthesized frames under both $N=2$ and $N=3$. Such even distribution of synthesized frames helps against over-fitting due to accumulation of samples at a particular moment.

Now we need a new index to measure the change of imbalance level during resampling. The global imbalance degree is not suitable here, since the performance of statistical moving object detection mostly depends on how many samples are involved in modeling of areas occupied by foreground. We define a new imbalance measure as $\eta = \tau_f/\tau_b$. In this way the numbers of foreground and background samples in potential foreground areas are roughly reflected by model lengths τ_f and τ_b. After temporal over-sampling, the frame number of foreground frames has become $N\tau_f$, and accordingly the imbalance degree becomes $\eta = (N\tau_f)/\tau_b$.

2.2. Spatial Over-sampling

Spatial over-sampling generates new samples for each synthesized frame. This process consists of two parts: warp samples from $\mathbf{F}^{t-s}$ to the corresponding instant $r(n)$, and then synthesize new samples by resampling the warped pixels. Motion information is necessary for warping. However, we cannot directly obtain motion field from $t-s$ to $r(n)$. Our solution is tracking foreground from $t-4$ to t, and then use such trajectories to predict motion vectors from $t-s$ to $r(n)$.

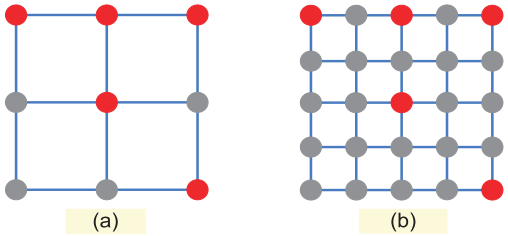

Fig. 5. A synthesized frame with different M. Red circles represent warped pixels; gray circles indicate they are empty grid nodes, and no samples are warped to these positions. (a): $M=1$. (b): $M=2$.

First we track foreground from $t-4$ to t based on the GPU-based point tracker [14]. This step divides each foreground frame $\mathbf{F}^{t-s}$ into two subsets, F_1 and F_2, which contain correctly and falsely tracked pixels, respectively. A trajectory of a foreground pixel is shown in Fig. 4a. Taking $\mathbf{F}^{t-4}$ and $N=3$ as an example, before warping we need compute the motion vectors of the pixel in Fig. 4a from instant $t-4$ to $r(1)$, $r(2)$ and $r(3)$. For each $r(n)$, we take the motion vector from $t-4$ to instant of integer closest to $r(n)$ to predict motion vector to $r(n)$. The closest instant of integer to $r(1)$ is $t-2$, so that we compute the motion vector to $r(1)$ based on the motion vector to $t-2$ (Fig. 4b). There are two closet instants to $r(2)$. Because $t-2$ has been used for $r(1)$, we compute the motion vector to $r(2)$ based on the motion vector to $t-1$ (Fig. 4c). The motion vector to $r(3)$ is also calculated using the above rules (Fig. 4d).

Let (v_{x_i}, v_{y_i}) be the motion vector of a correctly tracked pixel $\mathbf{z}_i$ from instant $t-s$ to $r(n)$. Assuming that this sample is warped to instant $r(n)$ and forms a new pixel $\mathbf{z}_j$. The horizontal ordinate of $\mathbf{z}_j$ is computed as

$$x_j = v_{x_i} + x_i. \tag{3}$$

y_j is computed with the same manner as x_j. Next we take spatial correlation to warp pixels in F_2. Assume $\mathbf{z}_i$ is a pixel from F_2 and its corresponding pixel at instant $r(n)$ is $\mathbf{z}_j$; $\mathbf{z}_k$ is the nearest neighbor of $\mathbf{z}_i$ in F_1. Our solution is keeping the relative position between $\mathbf{z}_i$ and $\mathbf{z}_k$ unchanged from $t-s$ to $r(n)$. The abscissa of $\mathbf{z}_j$ now can be computed as

$$x_j = v_{x_k} + x_i - x_k, \tag{4}$$

y_j is computed with the same manner as x_j. Eq. 4 does not guarantee the absolute precision, especially for non-rigid foreground. However, we can expect that most pixels in rigid foreground areas can be rightly warped.

Next new samples are created by resampling the warped pixels. The number of synthesized samples is controlled by

an integer $M, M > 0$. M is the resolution of the synthesized frames. $M = 1$ means the resolution of a synthesized frame is the same as the resolution of $\mathbf{Z}^t$. $M > 1$ means the resolution of a synthesized frame is increased M times. The advantage of resampling on a higher resolution is the spatial correlation among neighbors helps synthesize representative samples.

A synthesized frame with different M are depicted in Fig. 5. It can be seen that there are more empty nodes with a larger M. New samples will be synthesized at certain such empty nodes. we use a mask convolution for this purpose. A mask is a $(2\varepsilon M-1)*(2\varepsilon M-1)$ square matrix, and has the same resolution with the synthesized frame:

$$\mathcal{M}(i,j) = \frac{1}{\mathcal{Q}_1}\mathcal{N}\big(\frac{i-\mu_1}{\sigma_1}, \frac{j-\mu_2}{\sigma_2}\big) \quad i,j = -\varepsilon M, ..., \varepsilon M \tag{5}$$

where ε controls the size of mask, $\mathcal{Q}_1$ is a normalization factor, and $\mathcal{N}$ is a 2D normal distribution: $\mathcal{N}(\mu_1, \mu_2, \sigma_1, \sigma_2, \varrho)$. Here we set $\mu_1 = \mu_2 = 0$, $\sigma_1 = \sigma_2 = 1$ and $\varrho = 1$. Define a flag image G with the same resolution as the synthesized frame, where $G(x,y) = 0$ for nodes occupied by warped pixels, and 0 for empty nodes. For an empty note at (x,y), a new sample $\mathbf{z}(x,y)$ is synthesized by mask convolution as

$$\mathbf{z}(x,y) = \frac{1}{\mathcal{Q}_2}\sum_{i=-M}^{M}\sum_{j=-M}^{M} G(x+i,y+j)\mathbf{z}(x+i,y+j)\mathcal{M}(i,j) \tag{6}$$

where $\mathcal{Q}_2$ is another normalization factor: $\mathcal{Q}_2 = \sum_i\sum_j G(x+i,y+j)\mathcal{M}(i,j),\ i,j = -\varepsilon M, ..., \varepsilon M$.

This mask convolution can be explained as follows. If some of the warped pixels fall into the area covered by mask centered at an empty note, a new sample is created as the weighted sum of the covered warped pixels. The weights are proportional to the distances between the warped pixels and the empty grid. The number of warped pixels that are covered is decided by ε. No samples are created if none of the warped pixels fall into the area covered by mask.

Considering the influence of detection noise, a 'clean' operation is conducted on $\mathbf{F}^{t-s}$ before resampling. In addition, we use a mixture $\{\mathbf{F}^{t-s}\} \cup \{\tilde{\mathbf{F}}^{r(n)}\}$ as the final foreground training set to compensate possible resampling errors. Compared with $\mathbf{F}^{t-s}$, each corresponding synthesized frame is enlarge M^2 times. That's to say, the number of samples in a synthesized frame now is about M^2 times of that of a reference frame. Accordingly, the imbalance degree becomes

$$\eta = \frac{(NM^2+1)\tau_f}{\tau_b}. \tag{7}$$

3. MODELING AND CLASSIFICATION

We use the non-parametric method to model both foreground and background. $\{\mathbf{F}^{t-s}\} \cup \{\tilde{\mathbf{F}}^{r(n)}\}$ and the last τ_b background frames are directly used as foreground and background models. The membership probabilities are computed with foreground and background models, and is then used to assign each pixel $\mathbf{z}_i$ a label $l_i \in (0,1)$, where 0 and 1 represent background and foreground, respectively. The probability of a new-coming pixel $\mathbf{z}_i$ belonging to foreground is computed using kernel density estimation (KDE) as

$$p(\mathbf{z}_i|l_i = 1) = J^{-1}\sum_j \varphi_H(\|\mathbf{z}_i - \mathbf{f}_j\|) \tag{8}$$

where $\mathbf{f}_j$ is a sample in the non-parametric foreground model, and J is the total number of samples involved in the computation of Eq. 8. φ is a kernel function and H is the width of kernel. The background probability $p(\mathbf{z}_i|l_i = 0)$ is computed with the same manner as Eq. 8.

The probability regularisation allows enforcement of smoothness into classification. The smoothness constraint is modeled with the first-order Markov Random Filed as

$$\phi_{i,j}(\mathbf{z}_i, \mathbf{z}_j) = l_i l_j + (1-l_i)(1-l_j), \quad \text{if } (\mathbf{z}_i, \mathbf{z}_j) \in \psi \tag{9}$$

where ψ is the set of 8-neighboring pixel pairs. Final classification is achieved by minimizing the energy function as

$$E = \sum_i \ln(\phi_i) l_i + \lambda \sum_i\sum_j \phi_{i,j} \tag{10}$$

where λ is a weighting factor for the balance between the smoothness term $\phi_{i,j}$ and the data term ϕ_i. ϕ_i is defined as the likelihood ratio of membership probabilities

$$\phi_i(l_i) = \frac{p(\mathbf{z}_i|l_i = 1)}{p(\mathbf{z}_i|l_i = 0)}. \tag{11}$$

We chose the graph cut for the regularisation of Eq. 10. Graph cut also has become something of a standard in background subtraction, so please refer to [16] for the use of graph cut in moving object detection.

4. EXPERIMENTS AND DISCUSSIONS

In this section we demonstrate parameter settings and comparison experiments. Due to limitation of space, we only discuss the setting of one parameter and donot describe each contrast algorithm in detail. Sequences in CDW12 database and six algorithms are chosen for test and comparison.

1) Selection of parameters: We only demonstrate the selection process of the imbalance degree d, which equals to the selection of N and M. Naturally TN_{rate} and TP_{rate} are suitable for parameter selection. However, TN_{rate} is not sensitive to false alarms. We chose TP_{rate} in conjunction with *PR* and *F-measure* (FM) as evaluation indexes:

$$PR = \frac{TP}{TP+FP} \quad \text{and} \quad FM = \frac{2 * TP_{rate} * PR}{TP_{rate} + PR}.$$

PR is able to reflect similar information to TN_{rate} but more sensitive to false alarms. Higher *PR* means higher TN_{rate} and

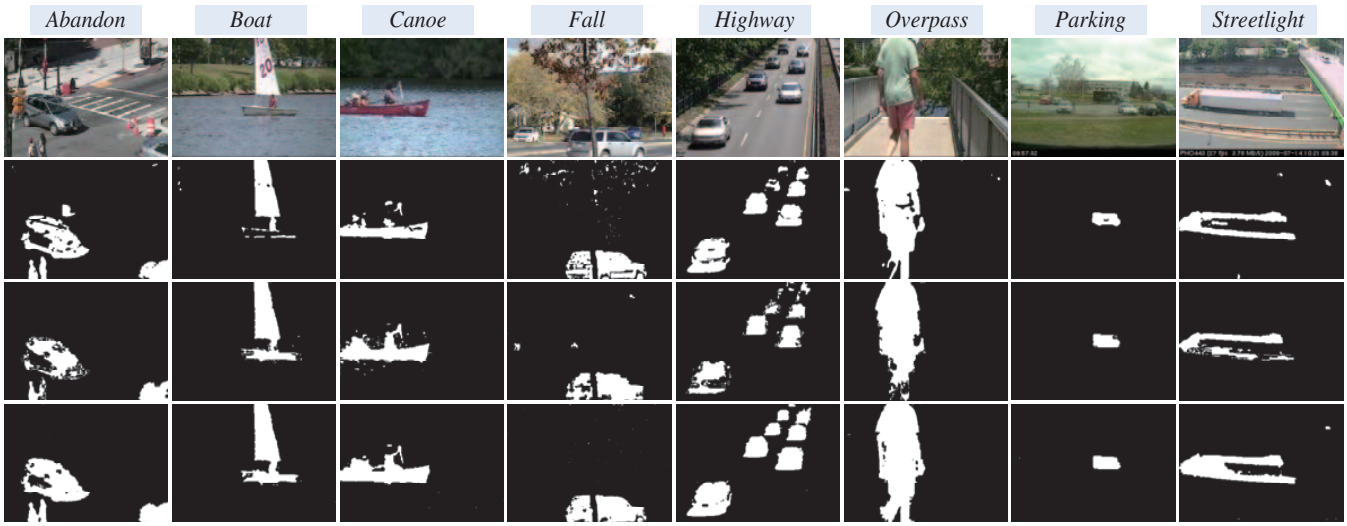

Fig. 6. Experiments by SGMM [10], CHEBY [8] and our method on eight sequences, are shown from the second to the last rows.

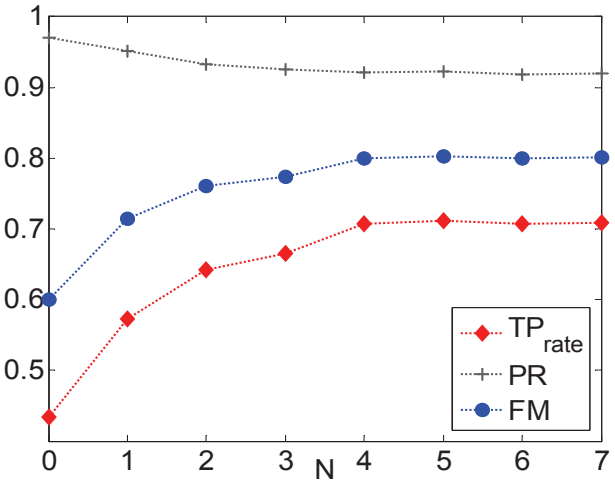

Fig. 7. Performance curves (TP_{rate}, *PR*, *FM*) of spatio-temporal over-sampling with variable N and fixed M.

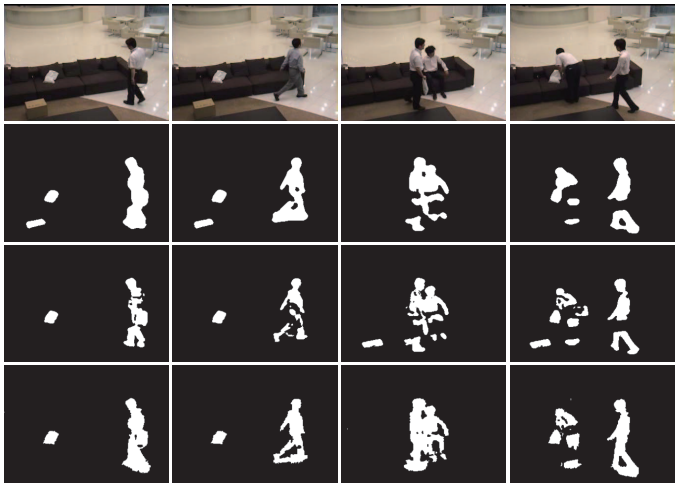

Fig. 8. Experiments by PBAS [4], SOBS [10] and our method on the *Sofa* sequence, are shown from the second to the last rows.

less false alarms. Higher *FM* can be achieved by a method only if it supports both higher TP_{rate} and higher *PR*.

First our method is performed on all selected sequences with variable N and fixed M. We set $M = 3$ and change N from 0 to 7, and performance curves are depicted in Fig. 7. It can be found that the improvement of TP_{rate} is accompanied by the degradation of *PR*. However, the consecutive growth of *FM* indicates that such degradation of *PR* is controllable. Furthermore, we find that the change of all the three indexes are no longer significant with $N > 5$. This phenomenon is known as over-fitting in data mining [15], where newly generated samples cannot further improve the representation ability of the minority training sample set. Next M is tested as N. Finally, to minimize the false alarms, we fix $N = 3$, $M = 2$ and $\tau_b = 50$ in all experiments, which equals $d = 1.04$. Such an imbalance level means the foreground and background sample sets are successfully rebalanced, which will certainly to bring performance improvement of detection.

2) Comparison with state-of-the-arts: Algorithms on CDW12 webpage are chosen for comparison. First, PBAS [4], SOBS [10] and our method are conducted on the *Sofa* sequence, as shown in Fig. 8. Compared with the two methods, performance is improved by our method mainly in the camouflaged areas. Reasons for such phenomenon is as follows. Camouflage means foreground and background in the same spatial vicinity share similar color distributions, resulting misclassification of foreground as background. Those misclassified foreground pixels are then used for background modeling and classification in the next frame, leading to further misclassification of foreground samples. Consequently, the imbalance degree of statistical approaches is enhanced. That is to say, camouflage is one reason for the class imbalance in background subtraction, and hence our method is also effective in resolving camouflage. In addition, Fig. 8 reveals that false alarms due to shadow is enhanced by our method. This is one reason for the decrease of *PR* in Fig. 7.

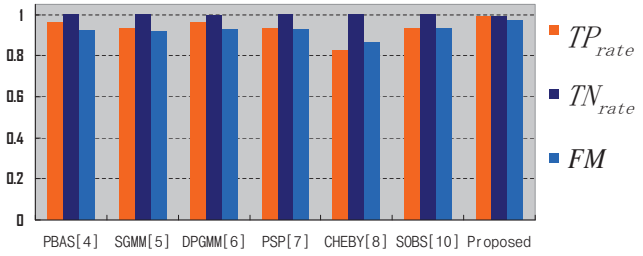

Fig. 9. Quantitative results on subset1 of the CDW12 database.

Next, SGMM [5], CHEBY [8] and our method are conducted on eight sequences, as shown in Fig. 6. The eight sequences cover a wide range of challenges and include various types of moving objects. It can be seen that performance is improved on almost all sequences by our method, which proves the existence of class imbalance and the effectiveness of the proposed method. Furthermore, we find that perfect results cannot be achieved by any method, which indicates that the low TP_{rate} in Fig. 1 is not caused by only class imbalance, but the joint result of multiple factors including class imbalance, dynamic background, etc.

In addition, two closely related methods are also tested for further performance evaluation: the Bayesian modeling method (BM) in [16] and the proposed method without graph cut (WGC). BM has the same non-parametric modeling and graph cut as our method, but without imbalanced learning. Averaged *FM* of BM, WGC and our method on sequences in Fig. 6 are 0.63, 0.87 and 0.89, respectively. Visual observation reveals that the improvement of our method over BM is mainly in the camouflaged areas, and scattered segmentation noises significantly increase by WGC, which is due to the omitting of the smoothness constraint in graph cut.

Finally, quantitative results of six high-ranked algorithms on CDW12 webpage and our method are computed based on the first subset of CDW12 database, as shown in Fig. 9. It can be clearly seen that our method generates the best TP_{rate} and *FM*. All above experiments also reveal that the main advantage of our method is the heavy computation load of KDE. Real-time speed is expected in the future with better hardware devices and algorithm optimization.

5. CONCLUSIONS

The class imbalance problem in moving object detection is studied in this work. First the imbalance nature in background subtraction is discussed, and then a spatio-temporal over-sampling method is presented to resolve the class imbalance in background subtraction from data level. Experiments are conducted on typical sequences to demonstrate the performance of our method. Future works include accelerating our method and introducing more imbalanced learning methods into moving object detection, e.g., kernel based solutions.

6. REFERENCES

[1] K. Toyama, J. Krumm, B. Brumitt and B. Meyers, "Wallflower: Principles and practice of background maintenance," in *ICCV*, Sep. 1999, pp. 255-261.

[2] S. Brutzer, B. Hoferlin and G. Heidemann, "Evaluation of background subtraction techniques for video surveillance," in *ICCV*, Nov. 2011, pp. 1937-1944.

[3] H. He and E.A. Garcia, "Learning from imbalanced data." *TKDE*, vol. 1, no. 9, pp. 1263-1284, Sep. 2009.

[4] M. Hofmann, P. Tiefenbacher, G. Rigoll, "Background segmentation with feedback: the pixel-based adaptive segmenter," in *Proc. IEEE Workshop on Change Detection*, Jun. 2012, pp. 38-43.

[5] R.H. Evangelio and T. Sikora, "Complementary background models for the detection of static and moving objects in crowded environments," in *Proc. IEEE Int. Conf. Advanced Video and Signal Based Surveillance*, Aug. 2012, pp. 71-76.

[6] T.S.F. Haines and T. Xiang, "Background subtraction with dirichlet processes," in *ECCV*, 2012.

[7] A. Schick, M. Bauml, R. Stiefelhagen, "Improving foreground segmentations with probabilistic superpixel Markov random field," in *Proc. IEEE Workshop on Change Detection*, 2012.

[8] A. Morde, X. Ma, S. Guler, "Learning a background model for change detection," in *Proc. IEEE Workshop on Change Detection*, 2012.

[9] Z. Zivkovic and F.V.D. Heijden, "Efficient adaptive density estimation per image pixel for the task of background subtraction," *PRL*, vol. 27, no. 7, pp. 773-780, Jul. 2006.

[10] L. Maddalena and A. Petrosino, "A self-organizing approach to background subtraction for visual surveillance applications," *TIP*, vol. 17, no. 7, pp. 1168-1177, Jul. 2008.

[11] A. Elgammal, R. Duraiswami, D. Harwood and L. Davis, "Background and foreground modeling using non-parametric kernel density estimation for visual surveillance," *Proc. IEEE*, vol. 90, no. 7, pp. 1151-1163, Jul. 2002.

[12] G.M. Weiss, "Mining with rarity: a unifying framework," *SIGKDD Explorations Newsletter*, vol. 6, no. 1, pp. 7-19, Jan. 2004.

[13] B.C. Wallace, K. Small, C.E. Brodley, T.A. Trikalinos, "Class imbalance, redux," in *ICDM*, Dec. 2011, pp. 754-763.

[14] N. Sundaram, T. Brox and K. Keutzer, "Dense point trajectories by GPU-accelerated large displacement optical flow," in *ECCV*, Sept. 2010, pp. 438-451.

[15] H. He, Y. Ma, X.-Y. Liu, Z.-H. Zhou, etc, Imbalanced learning: foundatons, algorithms, and applications, Wiley-IEEE, 2013.

[16] Y. Sheikh and M. Shah, "Bayesian modeling of dynamic scenes for object detection," *TPAMI*, vol. 27, no. 11, pp. 1778-1792, Nov. 2005.