Crowd Detection with BinBoost Descriptors

Daniel Moraes, Maurício Perez
Institute of Computing, University of Campinas
Av. Albert Einstein, 1251. Campinas, SP – Brazil
daniel.b.moraes@gmail.com, mauriciolp84@gmail.com

I. Introduction

Surveillance equipments, like closed-circuit television (CCTV), are standard for most of the industrial and commercial facilities, and even for some residential buildings and houses. Streets also have a large number of cameras for monitoring and identifying events that need attention. An event that may not be the cause, but is probably a result of an event that need further attention, is the emergence of crowds.

Despite being easy for humans to identify a crowd of people on a scene, it is usually too costly to maintain someone supervising all the time. Instead, it is much more desirable to have a camera attached to a computer analysing the scene and whenever it detects some suspicious movement of crowds, a warning message is sent to someone responsible.

The work here proposed is to analyze the use of BinBoost [1] descriptor at crowd detecting with Bag-of-Visual-Words (BoVW) [2], [3] and machine learning applied at the frames of the videos. The choice for a binary descriptor is related to the better efficiency that is expected with this kind of descriptors while classifying if it is a crowd event or not, while BoVW integrated with machine learning have already been show as a good approach at classifying images.

The rest of this paper is organized as follows. In Section II we discuss related work for crowd detection. Section III shows the evaluation methodology used to compare BinBoost and SURF descriptor on the problem of crowd detection, while Section IV gives a brief overview of the results we achieved. Finally, Section V concludes the paper and points out some possible future research opportunities.

II. Related Work

Binary descriptors aims to reduce both the computation time and description size, and is usually much more compact and faster than the traditional floating-point descriptors. Such gains in efficiency and memory size do not come for free, since they usually do not achieve results as good as the ones that are not limited in generating binary descriptions.

Binboost [1] is a novel binary descriptor, based on the AdaBoost classifier. In summary, it is a combination of weak learners that generates a set of gradient-based image features. The authors show that this descriptor outperforms even the best binary descriptors, like BGM, ITQ-SIFT, and LDAHash. They also state that BinBoost comes near to the best floating-points descriptors, like SIFT [4].

Most of the previous work on crowd analysis has different goals of the one we present here, that is only to determine if a certain image contains or not a crowd of people (crowd detection). Liang, Zhu, and Wang [5], for example, aims at discovering the crowd flow orientation. They use a Lucas-Kanade optical flow with Hessian. Fradi and Dugelay [6] try to count the approximate number of people in a crowd scene. Basically, they use a background subtraction technique with a Gaussian Mixture Model to separate the people from the background and count them. Arandjelovi [7] is the only work we found for crowd detection. The author tries to detect crowd scenes using the SIFT descriptor with a pyramid of sliding windows. Then, they use a SVM to give a crowd-like measure of each patch.

We believe that the Bag of Visual Words (BoVW) model will overcome the need for background subtraction. Also, the BoVW algorithm tends to select the features that should be linked to crowds, counting them, this way removing the need to count the number of people in the scene.

III. Evaluation Methodology

This section discusses the methodology for comparing the BinBoost binary descriptor with the SURF descriptor for the problem of crowd detection.

A. Datasets

To evaluate the BinBoost and SURF descriptors in the crowd detection scenario, we used two datasets. The PETS2009 S1 public dataset[1] for crowd analysis and a set of images that we collected on the Google Images website.

The PETS2009 S1 dataset has three groups of frames: Background, City Center, and Regular Flow, which consist of sequences of frames from different views. The difference among the frames of these subsets is described below:

- Background: few people (or nobody) in the range of the camera;
- City Center: some people (usually, 5 to 10 people) in the range of the camera, but moving in different directions;
- Regular Flow: many people (more than 10) moving in similar directions.

Since our definition of crowd does not take into account the direction in which people are moving, we did not use the

[1] http://www.cvg.rdg.ac.uk/PETS2009

City Center subset. The Regular Flow subset was used to build the crowd class and the Background subset for the non-crowd class.

The Background subset has eight views (numbered from 1 to 8), which in turn have a sequence of frames. Since the frames from the same view tend to be very similar, we separated two views (views 4 and 8) only for testing. It is also important to note that this dataset consists of frames from a video, and frames near in time to each other tend to be very similar. To avoid having similar images in this dataset, we only used a subset (randomly selected) of the images from each view.

The Regular Flow subset has four views (the same first four views from the Background subset), and we selected one of them (view 4) only for testing. Again, for avoid having similar images in the dataset, we only used one frame in ten from each view.

We will call this set of images (collected from the PETS2009 S1 dataset) as the *Crowd Pets* dataset. Table I shows the train/test sizes of this dataset.

TABLE I
CROWD PETS DATASET (IMAGES FROM PETS2009 S1 DATASET).

	Crowd	Non-Crowd
Train	240	300
Test	80	100

To build our second dataset, we chose to collect images on the Google Images website. The crowd images have been collected searching by the "crowd" keyword, and the non-crowd images using the keywords "neighborhood", "park", and "city". We picked 330 images from the "crowd" search, and 110 from the other three, totalizing a dataset of size 660 (330 for crowd and 330 for non-crowd). We will call this set of images as the *Crowd Google Images* dataset. Since we have no obvious train/test split in this dataset, we decided to evaluate it with a cross-validation approach, instead of a fixed train/test split. We will discuss about the cross-validation later in this section.

B. Experimental Setup

Most computer vision methods attempt to solve image classification problems in three main layers: low-level, mid-level, and high-level. The low-level is usually composed by some image processing and description methods. The mid-level is intended to analyse the low-level content and represent them in a more generalizable form. Many people say that this step aims to reduce the semantic gap between the data and the classification problem we want to solve. As for the high-level, it is usually composed by a learning algorithm that tries to distinct the classes of the problem through the feature vectors that are generated by the mid-level. Since our focus in this work is only to evaluate the BinBoost and SURF descriptors, we built a simple pipeline with basic mid- and high-level layers. Below, the methods we used on each of these layers are described in detail.

1) Low-level: Image description: In this layer, we did some image processing and used the BinBoost and SURF descriptors to describe the images. We started by downsampling the images to a maximum size of 480×360, keeping their original aspect ratio. Next, we converted the images to grayscale. These two steps were applied for both Crowd Pets and Crowd Google Images datasets, and we used the *mogrify* tool from the ImageMagick package. After that, we described the images using the aforementioned descriptors. We evaluated two approaches for selecting the image keypoints to be described with different descriptor sizes. The keypoints were selected using the original method inherent of each descriptor and alternatively in a dense form, with a patch of 24×24 and a sliding step of 4. As for the description sizes, we evaluated BinBoost with 128 and 256 dimensions and SURF with 64 dimensions. Table II shows the descriptors we used, with their respective settings.

TABLE II
DESCRIPTORS WE USED IN OUR EXPERIMENTS, WITH THEIR RESPECTIVE SETTINGS.

	Descriptor	Key-point selection	Size
BinBoost 128	BinBoost	BinBoost keypoints	128
BinBoost 256	BinBoost	BinBoost keypoints	256
BinBoost Dense 128	BinBoost	Dense, patch 24x24, step 4	128
BinBoost Dense 256	BinBoost	Dense, patch 24x24, step 4	256
SURF 64	SURF	SURF keypoints	64
SURF Dense 64	SURF	Dense, patch 24x24, step 4	64

2) Mid-level: Image representation: In the mid-level layer, we used the BoVW method. Despite simple, this technique has shown good results on many image classification problems. It starts by building a random sample of the data (usually 50% of the samples from the positive class and 50% from the negative class). This random sample is called "codebook". After that, the mid-level feature vectors are generated using coding and pooling methods. There are two main coding methods for the BoVW: hard and soft assignments. We chose to use the hard assignment, since it is simpler, faster and achieves comparable results. For the pooling, we used the average pooling.

3) High-level: Classification: In the classification phase we used the Support Vector Machine (SVM), a powerful algorithm for binary classification. We also used a Gaussian kernel (RBF), which is a good default kernel when we have no prior knowledge on how to represent the data under analysis.

In this case, the input of the SVM classifier are the feature vectors computed above with the BoVW technique. The parameters of the RBF SVM are selected in a grid-search fashion, using a 5-fold validation (on the training set) and selecting the pair of values with higher mean accuracy. Finally, these parameters are used to train a SVM model, which is then used to predict the samples we separated for testing.

IV. EXPERIMENTS AND RESULTS

In this section we compare the BinBoost descriptor with the SURF descriptor for the problem of Crowd detection. The results from each dataset is presented on tables containing the rates of True Positives (TP), False Positives (FP), True

Negatives (TN), False Negatives (FN), Accuracy (ACC) and the F1 score (also known as F-Measure). The F1-score is the harmonic mean of precision and can be computed as follows:

$$F_1 = \frac{2 \times \text{tp}}{2 \times \text{tp} + \text{fn} + \text{fp}} \quad (1)$$

A. Experiment on Crowd Pets dataset

Table III shows the results of this experiment, that used the Dataset 1. For the non-dense runs, BinBoost descriptor had bad performance, doing a little better than random classification, conversely SURF obtained a reasonably accuracy of almost 74%. Surprisingly, with dense approach the BinBoost descriptors greatly increased their performance, achieving an accuracy over 80% and outperforming the SURF results, that had a decrease, by more than 10%. As expected the BinBoost of 256 bits did better than the 128 length on both cases, but not by much.

TABLE III
RESULTS OF BINBOOST AND SURF ON THE CROWD PETS DATASET.

	TP	FP	TN	FN	ACC	F1
BinBoost 128	0.075	0.000	1.000	0.925	**58.88**	**0.140**
BinBoost 256	0.087	0.000	1.000	0.912	**59.44**	**0.161**
SURF 64	0.412	0.000	1.000	0.588	**73.87**	**0.584**
Dense BinBoost 128	0.988	0.320	0.680	0.013	**81.68**	**0.827**
Dense BinBoost 256	0.938	0.250	0.750	0.062	**83.35**	**0.833**
Dense SURF 64	0.325	0.000	1.000	0.675	**70.00**	**0.491**

On Table IV there is a comparison between the size on disk occupied by the descriptions.

TABLE IV
COMPARISON OF DISK SPACE BETWEEN BINBOOST AND SURF DESCRIPTORS.

Descriptor	Disk space
BinBoost 128	56M
BinBoost 256	94M
SURF 64	1.8G
Dense BinBoost 128	350M
Dense BinBoost 256	586M
Dense SURF 64	2.4G

B. Experiment on Crowd Google Images

On the next experiment we used the Crowd Google Images dataset. Table V contains its results, where the values in the parentheses inform the standard deviation of each measure. In this experiment there was a big jump on accuracy between non-dense and dense description. Still, the later had a slightly better performance, with all three descriptions having approximately the same results if we take into account the standard deviation. Once again, the binary descriptors unexpectedly classified correctly more images than SURF.

In the Figures 1, 2, 3, and 4 there are some examples of the wrong classified images on the Crowd Google Images dataset. There are some cases, like 3, in which we have a suspicion that the greatly repeated pattern all over the image has led the algorithm to fail.

Fig. 1. Crowd image, classified as non-crowd.

Fig. 2. Crowd image, classified as non-crowd.

Fig. 3. Non-crowd image, classified as crowd.

Fig. 4. Non-crowd image, classified as crowd.

TABLE V
RESULTS OF BINBOOST AND SURF ON THE CROWD GOOGLE IMAGE DATASET.

	TP	FP	TN	FN	ACC	F1
BinBoost 128	0.903 (0.01)	0.067 (0.01)	0.933 (0.01)	0.097 (0.01)	**91.80 (1)**	**0.917 (0.01)**
BinBoost 256	0.915 (0.02)	0.068 (0.02)	0.932 (0.02)	0.085 (0.02)	**92.35 (2)**	**0.923 (0.01)**
SURF 64	0.878 (0.02)	0.080 (0.03)	0.920 (0.03)	0.122 (0.02)	**89.90 (3)**	**0.897 (0.01)**
Dense BinBoost 128	0.932 (0.01)	0.051 (0.01)	0.949 (0.01)	0.068 (0.01)	**94.05 (1)**	**0.940 (0.01)**
Dense BinBoost 256	0.918 (0.02)	0.046 (0.01)	0.954 (0.01)	0.082 (0.02)	**93.60 (2)**	**0.934 (0.01)**
Dense SURF 64	0.897 (0.02)	0.067 (0.01)	0.933 (0.01)	0.103 (0.02)	**91.50 (2)**	**0.914 (0.01)**

On Table VI there is a comparison between the size on disk occupied by the descriptions of the Crowd Google Images dataset.

TABLE VI
COMPARISON OF DISK SPACE BETWEEN BINBOOST AND SURF DESCRIPTORS.

Descriptor	Disk space
BinBoost 128	63M
BinBoost 256	105M
SURF 64	389M
Dense BinBoost 128	274M
Dense BinBoost 256	460M
Dense SURF 64	1.9G

C. Experiment with a cross-validation scenario

In this Experiment we used the Crowd Google Images for training and Crowd Pets testing. In this case, using all the aforementioned descriptors, the SVM classified the whole testing set as non-crowd. We believe this happened because of the dissimilarity between the different type of crowds that each one contains. The Crowd Google Pets dataset has a much more sparse crowd, while on Crowd Google Images there is a much more denser version.

V. CONCLUSIONS

In this work, we compared the effectiveness of the BinBoost and SURF descriptors on the problem of crowd detection. We evaluated them in two crowd datasets, where the first one was built with images from the PETS 2009 S1 dataset and the second one with images collected on the Google Images website. We used in our experimental pipeline a BoVW model as the mid-level layer and a RVF SVM classifier as the high-level layer. With this pipeline, BinBoost with dense features outperforms the SURF descriptor in both datasets, with a accuracy of 83.35% and 94.05% for the Crowd Pets and Crowd Google Images, respectivelly. Also, the pipeline with BinBoost 128 descriptions in a dense form achieved a low false negative rate of 1.3% on Crowd Pets and 6.8% on Crowd Google Images. With those results, we have an effective and efficient pipeline for crowd detection.

For future work, other binary descriptors can be tested in this scenario (e.g., BGM, ITQ-SIFT, LDAHash). We can also improve our mid-level using a BoVW with soft assignment or using a different image representation model. There is also a need for improving our dataset, with more representative images of crowd scenes. Finally, since the main problem of crowd detection is achieving low false negative rates (most crowd scenes need to be detected), a false negative learning method can also be evaluated in this scenario.

REFERENCES

[1] T. Trzcinski, M. Christoudias, P. Fua, and V. Lepetit, "Boosting binary keypoint descriptors," in *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*. IEEE, 2013, pp. 2874–2881.

[2] J. Sivic and A. Zisserman, "Video google: a text retrieval approach to object matching in videos," in *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, Oct 2003, pp. 1470–1477.

[3] G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray, "Visual categorization with bags of keypoints," in *Workshop on statistical learning in computer vision, ECCV*, vol. 1, no. 1-22, 2004, pp. 1–2.

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

[5] R. Liang, Y. Zhu, and H. Wang, "Counting crowd flow based on feature points," *Neurocomputing*, vol. 133, no. 0, pp. 377 – 384, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0925231214000630

[6] H. Fradi and J. Dugelay, "Low level crowd analysis using frame-wise normalized feature for people counting," in *Information Forensics and Security (WIFS), 2012 IEEE International Workshop on*, Dec 2012, pp. 246–251.

[7] O. Arandjelovi, "Crowd detection from still images," in *BMVC*, 2008.