Person Identification in Webcam Images: An Application of Semi-Supervised Learning

Maria-Florina Balcan NINAMF@CS.CMU.EDU
Avrim Blum AVRIM@CS.CMU.EDU
Patrick Pakyan Choi PAKYAN@CS.CMU.EDU
John Lafferty LAFFERTY@CS.CMU.EDU
Brian Pantano BPANTANO@ANDREW.CMU.EDU
Mugizi Robert Rwebangira RWEBA@CS.CMU.EDU
Xiaojin Zhu ZHUXJ@CS.CMU.EDU
School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213 USA

Abstract

An application of semi-supervised learning is made to the problem of person identification in low quality webcam images. Using a set of images of ten people collected over a period of four months, the person identification task is posed as a graph-based semi-supervised learning problem, where only a few training images are labeled. The importance of domain knowledge in graph construction is discussed, and experiments are presented that clearly show the advantage of semi-supervised learning over standard supervised learning. The data used in the study is available to the research community to encourage further investigation of this problem.

1. Introduction

The School of Computer Science at Carnegie Mellon University has a public lounge, where leftover pizza and other food items from various meetings converge, to the delight of students, staff, and faculty. To help monitor the presence of food in the lounge, a webcam, sometimes called the *FreeFoodCam*[1], is mounted in a coke machine and trained upon the table where food is placed. After being spotted on the webcam, the arrival of (almost) fresh free food is heralded with instant messages sent throughout the School.

The FreeFoodCam offers interesting opportunities for research in semi-supervised machine learning. This paper presents an investigation of the problem of person identification in this low quality video data, using webcam images of ten people that were collected over a period of several months. The results highlight the importance of domain knowledge in semi-supervised learning, and clearly demonstrate the advantages of using both labeled and unlabeled data over standard supervised learning.

In recent years, there has been a substantial amount of work exploring how best to incorporate unlabeled data into supervised learning (Zhu, 2005). Several semi-supervised learning approaches have been proposed for practical applications in different areas, such as information retrieval, text classification (Nigam et al., 1998), and bioinformatics (Weston et al., 2004; Shin et al., 2004). In the context of computer vision, several interesting results have been obtained for object detection. Levin et al. (2003) introduced a technique based on co-training (Blum & Mitchell, 1998) for fitting visual detectors in a way that requires only a small quantity of labeled data, using unlabeled data to improve performance over time. Rosenberg et al. (2005) present a semi-supervised approach to training object detection systems based on self-training, and perform extensive experiments with a state-of-the-art detector (Schneiderman & Kanade, 2002; Schneiderman, 2004a; Schneiderman, 2004b) demonstrating that a model trained in this manner can achieve results comparable to a model trained in the traditional manner using a much larger set of fully labeled data.

In this work, we describe a new application of semi-supervised learning to the problem of person identification in webcam images, where the video stream has a low frame rate, and the images are of low quality. Significantly, many of the images may have no face, as the person could be facing away from the camera. We discuss the creation of the

[1] `http://www.cs.cmu.edu/~coke`, Carnegie Mellon University internal

Appearing in *Proc. of the 22^{st} ICML Workshop on Learning with Partially Classified Training Data*, Bonn, Germany, 2005. Copyright 2005 by the author(s)/owner(s).



Figure 1. Four typical FreeFoodCam images.

dataset, and the formulation of the semi-supervised learning problem. The task of face recognition, of course, has an extensive literature; see (Zhao et al., 2003) for a survey. However, to the best of our knowledge, person identification in video data has not been previously attacked using semi-supervised learning methods. Relatively primitive image processing techniques are used in our work; we note that more sophisticated computer vision techniques can be easily incorporated into the framework, and should only improve the performance. But the spirit of our contribution is to argue that semi-supervised learning methods may be attractive as a complementary tool to advanced image processing. The data we have developed and that forms the basis for the experiments reported here will be made available to the research community.[2]

2. The FreeFoodCam Dataset

The dataset consists of 5254 images with one and only one person in it. Figure 1 shows four typical images from the data. The task is not trivial:

- The images of each person were captured on multiple days during a four month period. People changed clothes, hair styles, and one person even grew a beard. We simulate a video surveillance scenario where images for a group of people are manually labeled in a few beginning frames, and the people must be recognized on later days. Therefore we choose labeled data within the first day of a person's appearance, and test on the remaining images of the day and all other days. This is much more difficult than testing only on the same day, or allowing labeled data to come from all days.

- The FreeFoodCam is a low quality webcam. Each frame has 640×480 resolution so faces of far away people are small. The frame rate is a little over 0.5 frames per second, and lighting in the lounge is complex and changing.

- A person could turn their face away from the camera, and roughly one third of the images contain no face at all.

Since only a few images are labeled, and all of the test images are available, the task is a natural candidate for the application of semi-supervised learning techniques.

[2] Instructions for obtaining the dataset can be found at `http://www.cs.cmu.edu/~zhuxj/freefoodcam`.



date	10/24	11/13	1/6	1/14	1/20	1/21	1/27	
1	128			193			153	474
2	256				193			448
3	288			305				593
4	204					190		394
5	266	41		189		19		515
6	195	34	179				104	512
7	126	163	200	180	70	22	28	789
8	189	66	172	117		15		559
9	189	94	215	69		30	43	640
10			65	143	122			330
total	1841	398	831	1196	384	276	328	5254

Figure 2. Left: mean background image used for background subtraction. Right: breakdown of the 10 subjects by date.

2.1. Data Collection

We asked ten volunteers to appear in seven FreeFoodCam takes over four months. Not all participants could show up for every take. The FreeFoodCam is located in the Computer Science lounge, but we received a live camera feed in our office, and took images from the camera whenever a new frame was available.

In each take, the participants took turns entering the scene, walking around, and "acting naturally," for example by reading the newspaper or chatting with off-camera colleagues, for five to ten minutes per take. As a result, we collected images where the individuals have varying poses and are at a range of distances from the camera. We discarded all frames that were corrupted by electronic noise in the coke machine, or that contained more than one person in the scene. This latter constraint imposed was to make the task simple to specify as a first step; there is no reason that the methods we present below could not be extended to work with scenes containing multiple people.

2.2. Foreground Color Extraction

To accurately capture the color information of an individual in the image, based primarily on their clothing, we had to separate him or her from the background. As computer vision is not the focus of the work, we used only primitive image processing methods.

A simple background subtraction algorithm was used to find the foreground. We computed the per-pixel means and variances of red, green and blue channels from 294 background images. Figure 2 shows the mean background. Using the means and variances of the background, we obtained the foreground area in each image by thresholding. Pixels deviating more than three standard derivations from the mean were treated as foreground.

To improve the quality of the foreground color histogram, we processed the foreground area using morphological transforms (Jain, 1989). Further processing was required because the foreground derived from background subtraction often captured only part of the body and contained background areas. We first removed small islands in the foreground by applying the *open* operation with a 7 pixel-wide square. We then connected vertically-separated pixel blocks (such as head and lower torso) using the *close* operation with a 60-pixel-by-10-pixel rectangular block. Finally, we made sure the foreground contains the entire person by enlarging the foreground to include neighboring pixels by further *closing* the foreground with a disk of 20 pixels in radius. And because there is only one person in each image, we discarded all but the largest contiguous block of pixels in the processed foreground. Figure 3 shows some processed foreground images.

After this processing the foreground area is represented by a 100-dimensional vector, which consists of a 50-bin hue histogram, a 30-bin saturation histogram, and a 20-bin brightness histogram.

2.3. Face Image Extraction

The face of the person is stored as a small image, which is derived from the outputs of a face detector (Schneiderman 2004a; 2004b) . Note that this is *not* a face *recognizer* (a face recognizer was not used for this task). It simply detects the presence of frontal or profile faces, and outputs the estimated center and radius of the detected face. We took a square area around the center as the face image. If no face was detected, the face image is empty. Figure 4 shows a few face images as determined by the face detector.

2.4. Summary of the Dataset

In summary, the dataset is comprised of 5254 images for ten individuals, collected during seven takes over four months. There is a slight imbalance in the class distribu-



Figure 3. Examples of foregrounds extracted by background subtraction and morphological transforms.



Figure 4. Examples of face images detected by the face detector.

tion, and only a subset of individuals are present in each day (refer to Table 2 for the breakdown). Overall 34% of the images (1808 out of 5254) do not contain a face.

Each image in the dataset is represented by three features:

Time: The date and time the image was taken.

Color histogram of processed foreground: A 100 dimensional vector consisting of three histograms of the foreground pixels, a 50-bin hue histogram, a 30-bin saturation histogram, and a 20-bin brightness histogram.

Face image: A square color image of the face (if present). As mentioned above, this feature is missing in about 34% of the images.

3. The Graphs

Graph-based semi-supervised learning depends critically on the construction and quality of the graph. The graph should reflect domain knowledge through the similarity function that is used to assign edges (and their weights). For the FreeFoodCam data the nodes in the graph are the images. An edge is formed between two images according to the following criteria:

1. *Time edges.* People normally move around in the lounge at moderate speed, thus adjacent frames are likely to contain the same person. We represent this knowledge in the graph by putting an edge between two images if their time difference is less than a threshold t_1 (usually a few seconds).



image 2910 | neighbor 1: time edge | neighbor 2: color edge

neighbor 3: color edge | neighbor 4: color edge | neighbor 5: face edge

Figure 5. A random image and its neighbors in the graph.

2. *Color edges.* The color histogram is largely determined by a person's apparel. We assume people change clothes on different days, so that the color histogram tends to be unusable across multiple days. However, it is an informative feature during a shorter time period (t_2), such as half a day. In the graph for every image i, we find the set of images having a time difference between (t_1, t_2) to i, and connect i with its k_c-nearest neighbors (in terms of cosine similarity on histograms) in the set. The parameter k_c is a small integer, such as three.

3. *Face edges.* We use face similarity over longer time spans. For every image i with a face, we find the set of images more than t_2 apart from i, and connect i with its k_f-nearest neighbor in the set. We use pixel-wise Euclidean distance between face images, where the pair of face images is scaled to the same size.

The final graph is the union of the three kinds of edges. The edges are unweighted. We used $t_1 = 2$ seconds, $t_2 = 12$ hours, $k_c = 3$ and $k_f = 1$ below. Conveniently, these parameters result in a connected graph.

It is impossible to visualize the whole graph. Instead, we show the neighbors of a random node in Figure 5.

4. Algorithms

We use the simple Gaussian field and harmonic function algorithm (Zhu et al., 2003) on the FreeFoodCam dataset. Let l be the number of labeled images, u the number of unlabeled images, and $n = l + u$. The graph is represented the $n \times n$ weight matrix W. Let D be the diagonal degree matrix with $D_{ii} = \sum_j W_{ij}$, and define the combinatorial Laplacian

$$L = D - W \tag{1}$$

Let Y_l be an $l \times C$ label matrix, where $C = 10$ is the number of classes. For $i = 1 \dots l$, $Y_l(i, c) = 1$ if labeled image i is in class c, $Y_l(i, c) = 0$ otherwise. Then the harmonic function solution for the unlabeled data is

$$Y_u = -L_{uu}^{-1} L_{ul} Y_l \tag{2}$$

where L_{uu} is the submatrix of L on unlabeled nodes and so on. Each row of Y_u can be interpreted as the collection of posterior probabilities $p(y_i = c|Y_l)$ for $c = 1 \dots C$ and $i \in U$. Classification is carried out by finding the class with the maximal posterior in each row.

In (Zhu et al., 2003) it has also been shown that incorporating class proportion knowledge can be helpful. The proportion q_c of data with label c can be estimated from the labeled set. In particular, the class mass normalization (CMN) heuristic scales the posteriors to meet the proportions. That is, one finds a set of coefficients $a_1, \dots, a_C$ such that

$$a_1 \sum_{i \in U} Y_u(i, 1) \ : \ \cdots \ : \ a_C \sum_{i \in U} Y_u(i, C) = q_1 \ : \ \cdots \ : \ q_C \tag{3}$$



Figure 6. An example "gradient walk" on the graph. The walk starts from an unlabeled image, through assorted edges, and ends at a labeled image.

Classification of an unlabeled point i is achieved by finding $\text{argmax}_c a_c Y_u(i,c)$. In the experiments below we report the accuracy of both the harmonic function and CMN.

4.1. Gradient Walks on the Graph

The harmonic algorithm described above solves a set of linear equations so that the predicted label of each example is the average of the predicted labels of its unlabeled neighbors and the actual labels of its labeled neighbors. The "reasons" for the algorithm's predictions can (roughly) be visualized by performing a "gradient walk" starting from an unlabeled example i, always moving to the neighbor with the highest score given to the predicted label. That is, let y be the predicted label for i. If we are at node j, we will walk to j's neighbor node k if

$$k = \text{argmax}_{k' \sim j} Y_u(k', y) \tag{4}$$

The gradient walk continues until we reach a labeled example. Two gradient walk paths are shown in Figure 6 and Figure 7.

5. Experimental Results

We evaluated harmonic functions on the FreeFoodCam tasks. For each task we gradually increased the labeled set size systematically, performed 30 random trials for each labeled set size. In each trial we randomly sampled a labeled set with the specified size from *the first day of a person's appearance only*. This is because we wanted to simulate a video surveillance scenario, where people are tagged and identified on later days. It is more difficult and more realistic than sampling labeled data from the entire dataset. If a class was missing from the sampled labeled set, we redid the random sampling. The remaining images are used as the unlabeled set.

We report the classification accuracies with harmonic functions and CMN, on two different graphs. The first graph is constructed with parameters $t_1 = 2$ seconds, $t_2 = 12$ hours, $k_c = 3$, $k_f = 1$, the second with $k_c = 1$. The results are presented in Figure 8.

To compare the graph-base semi-supervised learning methods against a standard supervised learning method, we used a Matlab implementation of support vector machines (Gunn, 1997) as the baseline. For C-class multiclass problems, we used a one-against-all scheme which creates C binary subproblems, one for each class against all the other classes, and select the class with the largest margin. Because we have missing features on face subimages, the kernel for the SVM baseline requires special care. We used an interpolated linear kernel $K(i,j) = w_t K_t(i,j) + w_c K_c(i,j) + w_f K_f(i,j)$, where K_t, K_c, K_f are linear kernels (inner products) on time stamp, color histogram, and face sub-image (normalized to 50×50 pixels) respectively. If image i contains no face, we define $K_f(i,\cdot) = 0$. The interpolation weights w_t, w_c, w_f were optimized with cross validation. Notice the SVMs with such kernel are not semi-supervised: the unlabeled data are merely used as test data. We found that the harmonic



Figure 7. An example "gradient walk" on the graph. The walk starts from an unlabeled image, through assorted edges, and ends at a labeled image.

function outperforms the linear kernel SVM baseline (Figure 8). The accuracy can be improved if we incorporate class proportion knowledge with the simple CMN heuristic. The class proportion is estimated from labeled data with Laplace (add one) smoothing.

To demonstrate the importance of using unlabeled data for semi-supervised learning, we compare the harmonic function with a minimal unlabeled set of size one. That is, for each unlabeled point x_i, we remove all other unlabeled points and compute the harmonic function on the labeled data plus x_i. This becomes a supervised learning problem. The harmonic solution with only one unlabeled point is equivalent to the standard weighted nearest neighbor algorithm. Since the original graphs are sparse, most unlabeled points may not have any labeled neighbors. To deal with this we instead connect x_i to its k_c nearest labeled neighbors in the color feature, and k_f nearest neighbors in the face feature, where edges are all unweighted. We tried various combinations of k_c and k_f, including those used in previous experiments. Notice we didn't use time edge at all, because it does not make sense with only one unlabeled point. The results are shown in Figure 9(a) with several different setting of k_c and k_f. The accuracies are all very low. Basically this shows that no combination of color and face works if one only use the labeled data. Therefore we see that using all the unlabeled data is quite important for our semi-supervised learning approach.

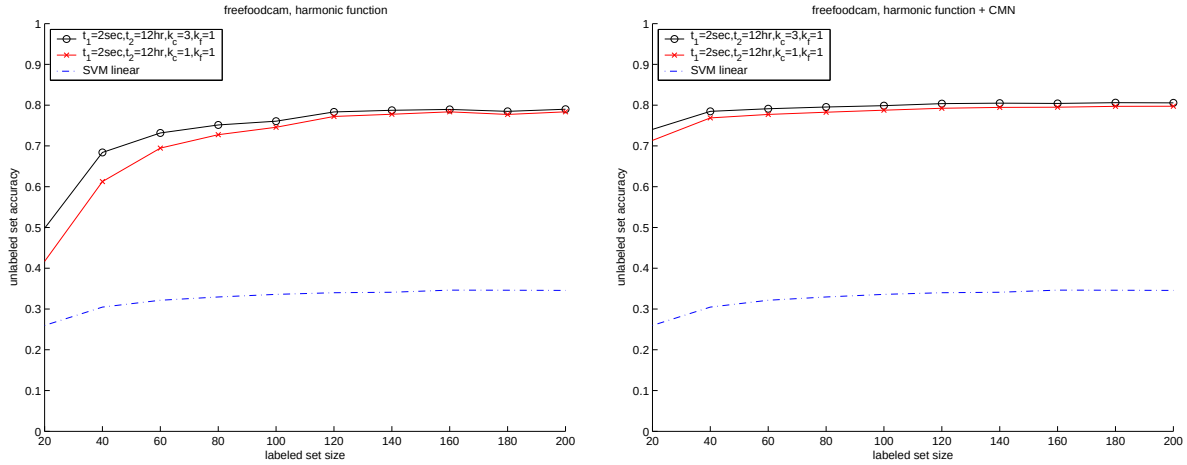
We assigned all the edges equal weights. A natural extension is to give certain types of edges more weight than others: e.g., perhaps give time-edges more weight than color-edges. In this case, rather than predicting each example to be the unweighted average of its neighbors, the prediction becomes a weighted average. Figure 9(b) shows that by setting weights judiciously (in particular, giving more emphasis to time-edges) one can substantially improve performance, especially on smaller samples. A related problem is to learn the parameters for K-nearest neighbor, i.e. k_c for color edges and k_f for face edges. We are currently exploring methods for learning good graph parameter settings from a small set of labeled samples.

6. Summary

In this paper we formulated a person identification task in low quality web cam images as a semi-supervised learning problem, and presented experimental results. The experimental setup resembles a video surveillance scenario: low image quality and frame rate; labeled data is scarce and is only available on the first day of a person's appearance; facial information is not always available in the image. Our experiments demonstrate that the semi-supervised learning algorithms based on harmonic functions are capable of utilizing the unlabeled data to identify ten individuals with greater than 80% accuracy. The dataset is now available to the research community.

Acknowledgements

We thank Henry Schneiderman for his help with the face detector, and Ralph Gross for helpful discussions on image processing. We also thank the volunteers for partici-



(a) harmonic function accuracy

(b) harmonic function + CMN accuracy

Figure 8. Harmonic function and CMN accuracy on two graphs. Also shown is the SVM linear kernel baseline. (a) The harmonic function algorithm significantly outperforms the linear kernel SVM, demonstrating that the semi-supervised learning algorithm successfully utilizes the unlabeled data to associate people in images with their identities. (b) The semi-supervised learning algorithm classifies even more accurately by incorporating class proportion knowledge through the CMN heuristic.

pating in the FreeFoodCam dataset collection. This work was supported in part by the National Science Foundation under grants CCR-0122581 and IIS-0312814.

References

Blum, A., & Mitchell, T. (1998). Combining labeled and unlabeled data with co-training. *COLT: Proceedings of the Workshop on Computational Learning Theory*.

Gunn, S. R. (1997). *Support vector machines for classification and regression* (Technical Report). Image Speech and Intelligent Systems Research Group, University of Southampton.

Jain, A. K. (1989). *Fundamentals of digital image processing*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc.

Levin, A., Viola, P. A., & Freund, Y. (2003). Unsupervised improvement of visual detectors using co-training. *International Conference on Computer Vision* (pp. 626–633).

Nigam, K., McCallum, A. K., Thrun, S., & Mitchell, T. M. (1998). Learning to classify text from labeled and unlabeled documents. *AAAI-98, 15th Conference of the American Association for Artificial Intelligence* (pp. 792–799).

Rosenberg, C., Hebert, M., & Schneiderman, H. (2005). Semi-supervised self-training of object detection models. *Seventh IEEE Workshop on Applications of Computer Vision*.

Schneiderman, H. (2004a). Feature-centric evaluation for efficient cascaded object detection. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Schneiderman, H. (2004b). Learning a restricted Bayesian network for object detection. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Schneiderman, H., & Kanade, T. (2002). Object detection using the statistics of parts. *International Journal of Computer Vision*.

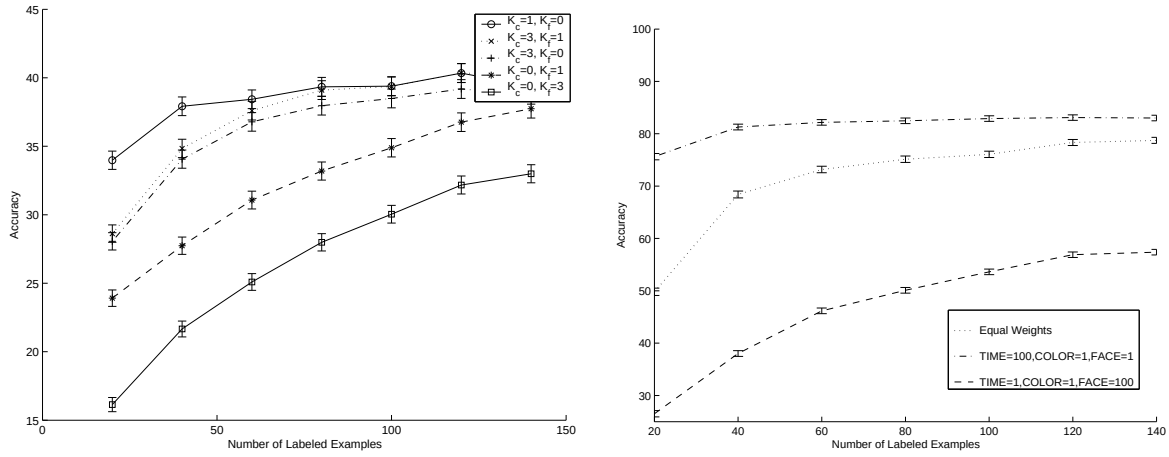
Shin, H., Tsuda, K., & Schlkopf, B. (2004). Protein functional class prediction with a combined graph. *Proceedings of the Korean Data Mining Conference* (pp. 200–219).

Weston, J., Leslie, C., Zhou, D., Elisseeff, A., & Noble, W. S. (2004). Semi-supervised protein classification using cluster kernels. *Advances in Neural Information Processing Systems 16*.

Zhao, W., Chellappa, R., Phillips, P. J., & Rosenfeld, A. (2003). Face recognition: A literature survey. *ACM Computing Surveys*, *35*, 399–458.

Zhu, X. (2005). *Semi-supervised learning with graphs*. Doctoral dissertation, Carnegie Mellon University. CMU-LTI-05-192.

Zhu, X., Ghahramani, Z., & Lafferty, J. (2003). Semi-supervised learning using Gaussian fields and harmonic functions. *ICML-03, 20th International Conference on Machine Learning*.



(a) using one unlabeled point at a time

(b) using different weights

Figure 9. (a) Using only one unlabeled point at a time in harmonic function. This is much worse than using all the unlabeled data for semi-supervised learning. (b) A comparison of three weighting schemes on the time, color, and face edges. The graph weights have a significant effect on the performance of the harmonic function algorithm. By giving more weight to the time edges, the harmonic function algorithm performs substantially better.