

# Boosting Chamfer Matching by Learning Chamfer Distance Normalization

Tianyang Ma, Xingwei Yang, and Longin Jan Latecki

Dept. of Computer and Information Sciences, Temple University, Philadelphia.  
{tianyang.ma, xingwei, latecki}@temple.edu

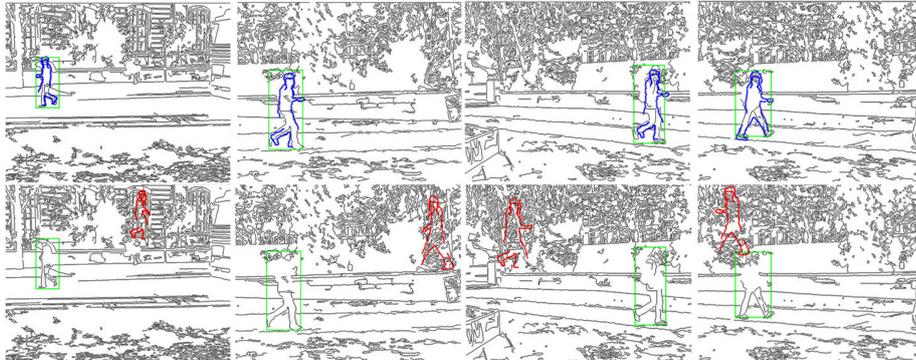
**Abstract.** We propose a novel technique that significantly improves the performance of oriented chamfer matching on images with cluttered background. Different to other matching methods, which only measures how well a template fits to an edge map, we evaluate the score of the template in comparison to auxiliary contours, which we call normalizers. We utilize AdaBoost to learn a Normalized Oriented Chamfer Distance (NOCD). Our experimental results demonstrate that it boosts the detection rate of the oriented chamfer distance. The simplicity and ease of training of NOCD on a small number of training samples promise that it can replace chamfer distance and oriented chamfer distance in any template matching application.

## 1 Introduction

Chamfer matching has been widely used for edge based object detection and recognition in computer vision. However, its performance is seriously limited in cluttered images. One of the main drawbacks of chamfer matching is the fact that a given template often fits better to a cluttered background than to the location of a true target object. Oriented chamfer matching (OCD) [17] adds orientation information, which significantly improves the performance of chamfer matching, but the problem still remains, as illustrated in Fig. 1. The proposed approach provides a solution to this problem by comparing the matching score of the template to normalizers, which are curve segments of varying but simple shape. There are two key properties of the normalizers. (1) If the target template matches well to a cluttered background, then very likely some of the normalizers match well too. (2) If the template matches well to a true object location, it is very unlikely for any normalizer to match well. Consequently, the normalized oriented chamfer distance (NOCD) significantly improves the discriminative power of OCD. Some examples are shown in Fig. 1.

Since it is hard if not impossible to satisfy (1) and (2) with a finite set of normalizers for a given set of target templates, we treat normalized chamfer distances as weak classifiers and employ AdaBoost to learn their weights. The weights provide a soft way of selecting adequate normalizers for a given template. As our experimental results demonstrate, AdaBoost is able to learn the normalizer weights on a small set of training images, which makes the proposed

approach suitable for all practical applications currently based on (oriented) chamfer matching.



**Fig. 1.** Example detection results on 250 test images from TU Darmstadt Pedestrian Dataset. The first row shows the detection results of the proposed NOCD, while the second row shows oriented chamfer matching results. The green rectangle denotes the ground truth bounding box.

The paper is structured as follows. In Section 3, we review basic definitions of chamfer distance and oriented chamfer distance. The new concept of distance normalization is introduced in Section 4, and AdaBoost learning of their weights is described in Section 5. Section 6 describes a simple framework for object detection. Finally, Section 7 introduces our set of normalizers. The performance of our method is evaluated and compared to OCD in Section 8.

## 2 Related work

There is a large number of applications of chamfer matching in computer vision and in medical image analysis. Chamfer distance was first introduced by Barrow et al. [2] in 1977 with a goal of matching two collections of contour fragments. Until today chamfer matching is widely used in object detection and classification task due to its tolerance to misalignment in position, scale and rotation. Borgefors [16] introduced a modified chamfer matching method called hierarchical chamfer matching, which could be regarded as a coarse-to-fine process by matching edge points using a resolution pyramid of the image. This method focuses on alleviating the computational load for chamfer matching. Meanwhile, chamfer matching meets the real-time system requirement due to fast implementations of distance transforms. Gavrila and Munder [3] performed template matching based on chamfer distance transform as a core technique to construct a real-time detection system of pedestrians.

Leibe et al. [4] used chamfer matching to detect pedestrian in crowded scenes, and combined segmentation as a verification to prevent the false alarms that

mostly lie in the cluttered background. Stenger et al. [6] introduced a template hierarchy which is formed by bottom-up clustering based on the chamfer distance. In [7], Opelt et al. used chamfer distance to score each boundary fragment for selection of candidate contour fragments. Opelt et al. also compared each boundary fragment from each category to all existing alphabet entries using chamfer distance in [8]. Other methods that utilize chamfer distance as shape similarity metric include [9, 13, 20]. Chamfer distance plays also an important role in medical image analysis, e.g., [10–12].

However, methods that utilize chamfer distance to measure the similarity between the template and edge maps suffer from mismatching to the cluttered background. It is generally agreed that main negative effect of using chamfer distance is the potential risk of increasing false alarms occurring in background with high level of clutter noise. Thayananthan et al. [14] compared the localization performance of chamfer matching and shape context [15], and concluded that chamfer matching is more robust in clutter than shape context matching even though most failure cases in chamfer matching are still due to false positive matches.

Recently, Shotton et al. [17] proposed an oriented chamfer distance (OCD) that exploits edge orientation information in the form of edge gradients. OCD linearly combines chamfer distance and orientation difference between template points and their closest matches, which leads to reduction of mismatching cases to the noisy background. Trinh and Kimia [25] proposed Contour Chamfer Matching (CCM) to improve OCD. In this method, based on the observation that the accidental alignment between a contour and the image edges always forms a zig-zagging contour, after finding the corresponding points in edge map, another orientation for edge points is computed based on the new generated curve, and an additional term which is the difference in tangent direction is taken into account when computing the Contour Chamfer Distance.

Since proposed method is not designed specifically for oriented chamfer distance, it could be also used to boost the performance of any distance metric that aims to capture edge support for a model. In particular, it would be possible to apply the proposed method to Hausdorff distance and oriented Hausdorff distance proposed in [26, 27], which is also widely used in computer vision applications. However, in [17] experimental evidence is provide that OCD has better performance than Hausdorff distance.

### 3 Oriented Chamfer Distance (OCD)

In this section we define chamfer distance and oriented chamfer distance (OCD), which is a simple linear combination between distance and orientation terms.

**Chamfer Distance** Chamfer distance was first proposed in [2] as an evaluation of 2D asymmetric distance between two set of edge points. It is tolerant to slight shape distortion caused by shift in location, scale and rotation. Given a template  $T$  positioned at location  $x$  in an image  $I$  and a binary edge map  $E$  of the image

$I$ , the basic form of chamfer distance is calculated as

$$d_{cham}^{(T,E)}(x) = \frac{1}{|T|} \sum_{x_t \in T} \min_{x_e \in E} \|(x_t + x) - x_e\|_2, \quad (1)$$

where  $\|\cdot\|_2$  is  $l_2$  norm and  $|T|$  denotes number of points in template  $T$ . Chamfer distance can be efficiently computed as:

$$d_{cham}^{(T,E)}(x) = \frac{1}{|T|} \sum_{x_t \in T} DT_E(x_t + x), \quad (2)$$

where  $DT_E$  is a distance transform defined for every image point  $x \in I$  as

$$DT_E(x) = \min_{x_e \in E} \|x - x_e\|_2. \quad (3)$$

Meanwhile, in practice, distance transform is truncated to a constant  $\tau$  [17]:

$$DT_E^\tau(x) = \min(DT_E(x), \tau) \quad (4)$$

This reduces the negative effective due to missing edges in  $E$ , and allows normalization to a standard range  $[0, 1]$ :

$$d_{cham,\tau}^{(T,E)}(x) = \frac{1}{\tau|T|} \sum_{x_t \in T} DT_E^\tau(x_t + x). \quad (5)$$

**Oriented Chamfer Distance (OCD)** Shotton et al. [17] proposed an improved chamfer distance called oriented chamfer distance (OCD), which adds additional robustness by exploiting edge orientation information. To define it, we first need a notation of an argument of a distance transform (ADT) that gives the locations of a closest point.

$$ADT_E(x) = \arg \min_{x_e \in E} \|x - x_e\|_2. \quad (6)$$

To evaluate a mismatch in orientation, the difference in tangent directions is computed

$$d_{orient}^{(T,E)}(x) = \frac{2}{\pi|T|} \sum_{x_t \in T} |\phi(x_t) - \phi(ADT_E(x_t + x))|, \quad (7)$$

where  $\phi(x)$  denotes tangent direction at point  $x$  and ranges between zero and  $\pi$ .  $|\phi(x_1) - \phi(x_2)|$  gives the smallest circular difference between  $\phi(x_1)$  and  $\phi(x_2)$ . Using a simple linear combination between the distance and orientation terms, oriented chamfer distance is defined as

$$OCD_\lambda^{(T,E)}(x) = (1 - \lambda) \cdot d_{cham,\tau}^{(T,E)}(x) + \lambda \cdot d_{orient}^{(T,E)}(x). \quad (8)$$

For clarity, we will omit  $E$  and  $\lambda$  below when possible, and use  $OCD(T, x) = OCD_\lambda^{(T,E)}(x)$  to represent the oriented chamfer distance of template  $T$  at location  $x \in I$ .

## 4 Normalization of Oriented Chamfer Distance

Although oriented chamfer matching adds orientation term to avoid mismatching, cluttered background still may match much better to the template than the real object contours. The reason is that cluttered background offers a large variety of edge orientations, consequently, any shape has a large probability of a good oriented chamfer score. This suggests that we need to compare the score of the target template with scores of some random shapes. If both have good OCD score at a given location, then the template match is most likely to be accidental. Based on this insight, we introduce a normalizer as an auxiliary, random shape to evaluate how well the template matches to the edge map at a certain location. For a target template  $T$ , we propose to generate  $K$  normalizers, denoted by  $\mathcal{N} = \{\eta_k | k = 1, \dots, K\}$ . A procedure to generate normalizers is described in Section 7. Instead of only calculating  $OCD(T, x)$  at each location  $x$ , we also compute  $OCD(\eta_k, x)$ , and compare the ratios

$$R_k(T, x) = \frac{OCD(T, x)}{OCD(\eta_k, x)}. \quad (9)$$

We call  $R_k(T, x)$  a **normalized score**.

Now we provide some details about the role of normalizers in improving chamfer score. The analysis is divided into three qualitative cases that illustrate an intended correct behavior of the normalizers. In practice, not all normalizers will behave in this way, which is addressed in Section 5.

**Case 1:** At a correct location containing a target object in a given image,  $OCD(T, x)$  is small and  $OCD(\eta_k, x)$  is large, so that  $OCD(T, x) < OCD(\eta_k, x)$ . Consequently,  $R_k(T, x)$  will become comparatively smaller than  $OCD(T, x)$ , which better indicates a correct match.

**Case 2:** In a cluttered area in which the target object is not present, both  $OCD(T, x)$  and  $OCD(\eta_k, x)$  are small, but  $OCD(T, x) > OCD(\eta_k, x)$ , so  $R_k(T, x)$  will become comparatively larger than  $OCD(T, x)$ , which better indicates a wrong match.

**Case 3:** In an area that is neither cluttered nor contains the target object, both  $OCD(T, x)$  and  $OCD(\eta_k, x)$  are large, but  $OCD(T, x) > OCD(\eta_k, x)$ , so  $R_k(T, x)$  will become comparatively larger than  $OCD(T, x)$ , which better indicates a wrong match.

Cases 1 to 3 clearly demonstrate that normalizers increase the discriminate power of OCD. However, they are based on an assumption that we have an ideal set of normalizers  $\{\eta_k | k = 1, \dots, K\}$  behaving as described in cases 1 to 3. Even though it may not be possible to find normalizers satisfying cases 1 to 3 for a given template  $T$ , we propose to utilize machine learning methods to learn which normalizers yield correct scores  $R_k(T, x)$  for a given template  $T$ . For a given set of candidate normalizers, we use AdaBoost in Section 5 to learn the weights of normalized scores  $R_k(T, x)$ . Thus, we treat each normalized score as a weak classifier. The weights provide a soft selection of a set of normalizers with our intuition being that this selection best approximates the behavior described in cases 1 to 3.

## 5 Learning Normalized OCD with AdaBoost

The standard AdaBoost [18] allows us to select a set of normalizers by assigning weights to their normalized scores and to combine them as a weighted linear combination, which yields a more robust matching score. Given is a set of training images with positive and negative examples, i.e, a set of bounding boxes containing the target object and a set of bounding boxes without the target object. AdaBoost automatically learns the weight for each weak learner and combine them to form a strong learner [21, 22]. We use the ratios  $R_k(T, x)$  as weak learners for  $k = 1, \dots, K$ . To be precise, a weak learner is defined as

$$h_k(T, x) = \begin{cases} 1 & \text{for } R_k(T, x) < th_k \\ 0 & \text{for } otherwise. \end{cases} \quad (10)$$

In each iteration  $1, \dots, K$ , we search for a weak learner with the best detection performance on the training set. During the search, the optimal threshold  $th_k$  for each weak learner is chosen to minimize the misclassification error (ME). At each iteration of AdaBoost, each training example carries a classification weight. ME is defined as the sum of the classification weights of misclassified training examples (both positives and negatives). As the output we obtain a strong learner

$$H(T, x) = \sum_{k=1}^K w_k \cdot h_k(T, x) \quad (11)$$

In the AdaBoost terminology, the value of the strong learner indicates how likely a given image location  $x$  belongs to the class of template  $T$ . The larger the value the most likely this is the case. We propose to replace the oriented chamfer distance of  $T$  with the value of  $H(T, x)$ . We define a **Normalized Oriented Chamfer Distance** as  $NOCD(T, x) = H(T, x)$ . While OCD is a distance in that the smaller is OCD value the better, NOCD is a similarity measure, i.e., the larger the NOCD value, the most likely the target object is present at location  $x$ .

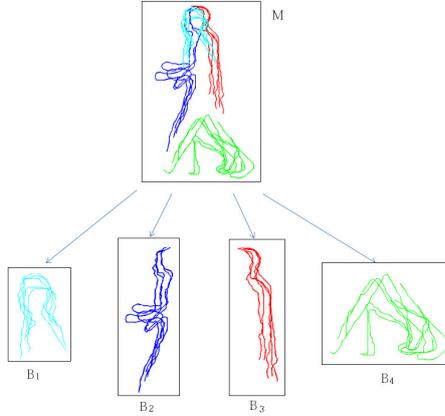
We use a simple strategy to select training examples for AdaBoost. Given is a set of training images with ground truth bounding boxes enclosing target objects. For each training image we select only 5 positive and 5 negative examples. As 5 positive examples we randomly select 5 locations in a small neighborhood around the ground truth locations. We select as negative examples 5 locations  $x$  with locally smallest oriented chamfer distance  $OCD(T, x)$  such that the area of the intersection of the bounding box centered at  $x$  with any ground truth bounding box is less than 50%.

## 6 Object Detection with NOCD

In order to be able to evaluate the performance of NOCD, we describe a very simple approach for object detection in this section. We keep it simple to allow

for clear comparison to OCD. However, we use a flexible shape model in our approach in order to be able to evaluate the performance of the proposed *NOCD* on state-of-the-art test datasets.

Our flexible object model is denoted as  $\mathcal{M} = \{B_i | i = 1, \dots, N\}$ , where  $B_i$  is a part bundle composed of contour parts describing the same location on the contour of a given shape class, e.g., human head or arm, and  $N$  is the number of bundles in model  $\mathcal{M}$ . Contour parts from bundle  $B_i$  are represented by  $c_{ij}$ , and hence  $B_i = \{c_{ij} | j = 1, \dots, M_i\}$ . Since every part bundle  $B_i$  describes a specific part of an object, we assume that  $B_i \cap B_j = \emptyset$  if  $i \neq j$ . Fig. 2 shows an example of human model, here  $N = 4$  and  $M_i = 5$  for  $i = 1, 2, 3, 4$ . Our model was manually constructed. Thus, our model contains the total of 20 contour parts  $c_{ij}$ . Each part  $c_{ij}$  is treated as template  $T$ , and  $NOCD(c_{ij}, x)$  is learned as describe in Section 5.



**Fig. 2.** Human model  $\mathcal{M}$  composed of 4 part bundles  $B_1, B_2, B_3, B_4$  representing head, front, back, and leg parts, respectively. Each bundle has 5 contour parts.

For an input image  $I$ , we first use Canny edge detector to compute the edge map  $E$ . For each location  $x$  in  $I$ , we use  $NOCD(c_{ij}, x)$  to represent the normalized oriented chamfer distance of model contour part  $c_{ij}$  placed at point  $x$ . With a simple but efficient sum-max framework, the model fit at point  $x \in I$  is defined as:

$$S_I(\mathcal{M}, x) = \sum_{i=1}^N \max_{c_{ij} \in B_i} NOCD(c_{ij}, x) . \quad (12)$$

Thus, we select from each bundle  $B_i$  the part with the largest NOCD score and sum the maximal scores over the bundles in the shape model  $\mathcal{M}$ . Using sliding window we calculate  $S_I(\mathcal{M}, x)$  at each point  $x \in I$ . We define the model fit score

as

$$S_I(\mathcal{M}) = \max_{x \in I} S_I(\mathcal{M}, x) \quad (13)$$

and the detection center point as point  $x^* \in I$  as

$$x^* = \arg \max_{x \in I} S_I(\mathcal{M}, x) \quad (14)$$

The detection results for *OCD* follow the same framework, but with max replaced with min in the above formulas.

## 7 Normalizers

It remains to describe how we select a set of normalizers  $\{\eta_k \mid k = 1, \dots, K\}$ . We first observe that a good normalizer should be more likely to match to noise than a given contour part. This implies that a normalizer should have a significantly simpler shape than the contour parts of a target shape model. We also want that a normalizer should be less likely to match to a true object edges in an image than a given contour part. Consequently, normalizers should not be similar to any contour parts in our shape models.



**Fig. 3. Basic normalizers.** Our set of basic normalizers contains 11 simple shapes.

We satisfy both constrains by first generating a small set of simple geometric curves that are treated as a basic structuring elements to generate a set of normalizers. A set of 11 basic shapes that we have selected is shown in Fig. 3. They form the first 11 elements of our set of normalizers  $\mathcal{N} = \{\eta_k \mid k = 1, \dots, K\}$ . We obtain further normalizers by pairwise combining the 11 structuring elements, where the combination is simply a union of their aligned images. Since the normalizer combination is symmetric and we only combine different structuring elements, we obtain  $55 = (11 \times 10)/2$  additional normalizers. Fig. 4 shows a complete set of  $K = 66$  normalizers obtained this way. They are ordered according to their weights obtained by the sum of AdaBoost weights of their corresponding weak classifiers by training the AdaBoost strong classifiers on the TU Darmstadt pedestrian dataset [1] (see Section 8 for more details). A larger weight indicate that a given normalizer makes more contribution in helping NOCD distinguish true positive from clutter background. The weight order of the normalizers confirms the simplicity principle that guided our design of

normalizers in that simpler normalizers are usually more significant. However, the weights of the normalizers are also influence by their ability to match well to noise, which may be image class specific. For example, straight lines in horizontal and vertical direction belong to a common background clutter in inner city images as the images of the TU Darmstadt pedestrian dataset.

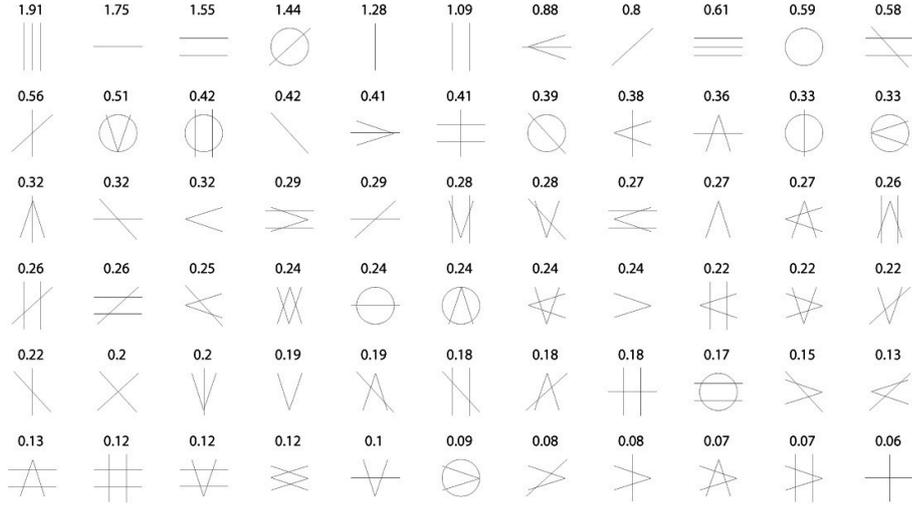
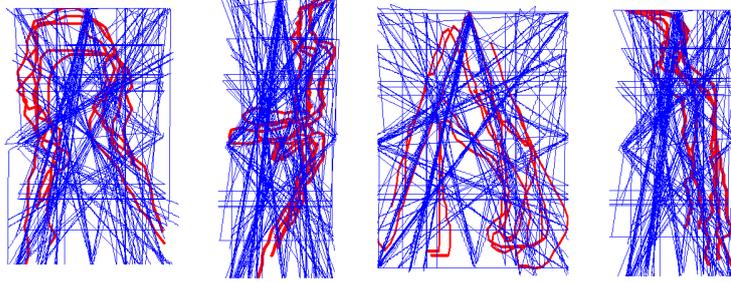


Fig. 4. Our 66 normalizers displayed in order of their weights.

For each contour part of a target model  $c_{ij}$ , we resize the normalizers to let them have the same bounding box as the contour part  $c_{ij}$ . Consequently, the resized normalizers cover the same area. Fig. 5 shows the resized normalizers generated for each bundle of the human model.

## 8 Experimental Evaluation of Detection Rate

In this section we compare object detection performance of the proposed normalized oriented chamfer distance (NOCD) to the oriented chamfer distance (OCD) and to chamfer distance on standard test datasets. The detection method is described in Section 6. We use exactly the same flexible models and the same experimental settings for both methods. In particular, for each image, the edge map was computed by the canny edge detector with the same threshold. The chamfer distance was computed exactly as defined in formula (5). The same constants  $\tau$  and  $\lambda$  were used to truncate the distance transform and linearly combine the distance and orientation terms when calculating the oriented chamfer distance. Results are quantified in terms of detection rate. We use the standard



**Fig. 5. Human model normalizers.** The resized normalizers for four part bundles are shown in blue. The red curves are the original model parts for each bundle.

PASCAL criterion to identify correct detections. A detection is regarded as correct if the area of the intersection of the bounding box containing the detected object with the ground truth bounding box is at least 50% of the area of their union.

*TU Darmstadt Pedestrian Dataset* Human detection is very challenging for shape-based matching methods, because in many poses the shape of human contours is relatively simple. In surveillance images, there is often a complex background, while humans are relatively small, which also increases the chance for an accidental matching.

TU Darmstadt pedestrian dataset [1] consists of several series of video images containing side-view humans. It provides two training datasets, one has 210 images and another has 400 images. In our experiment, we use training 400 dataset for the training of NOCD. After that, we test both NOCD and OCD on the test dataset with 250 images. The 250 test images are significantly more challenging than the 400 training images. To handle the variance of the human shape caused by people walking in opposite directions, we flip our model with respect to vertical axis, and take the best score of the original and flipped models. Consistent with the results of the  $\lambda$  learning procedure reported in Shotton [17], we also observed that detection accuracy of oriented chamfer distance increases when  $\lambda$  becomes larger. In all human detection experiments, we used  $\lambda = 0.8$  for both OCD and NOCD, which was the best performing. As it is often the case in AdaBoost applications, we discarded weak classifiers with very small weights. After training phase, we retained only 37 normalizers with largest weights to form the strong learner for each model contour part. This allows us to reduce the object detection cost complexity.

The detection rate is shown in Table 1. We observe that the proposed NOCD nearly doubled the detection rate of OCD on the 250 test images. The improvement is very significant given the fact that the detection rate of OCD is very low: 35.2%.

Several detection results are displayed in Fig. 1. As they illustrate OCD fails when the human contours are broken and distorted while at the same time the

Chamfer distance	4.4%	HOG [23]	72%
OCD	35.2%	4D-ISM [24]	81%
proposed NOCD	70%	Andriluka et al. [1]	92%

**Table 1.** Detection rate on Test 250 of the TU Darmstadt Pedestrian Dataset. The proposed NOCD doubled the OCD detection rate with exactly the same contour model.

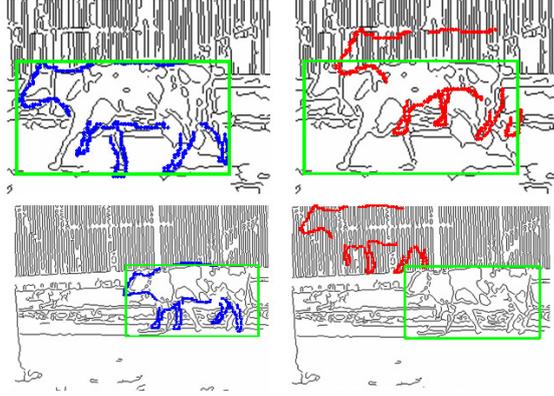
background is cluttered. This is exactly when the proposed NOCD performs extremely well. We also report the performance of pure chamfer distance in Table 1. in order to show that OCD performs significantly better than chamfer distance on this dataset. Further, we include the detection rates of state-of-the-art approaches estimated from graphs reported in [1]. We observe that our detection rate is compatible to a popular appearance based detector, HOG [23]. We stress that our approach is still a matching approach. Andriluka et al. [1] obtained the currently best performance on this dataset. It is obtained by an approach specifically designed for pedestrian detection that utilizes a sophisticated statistical inference framework and learning to handle articulations; both not present in our approach. Similarly, the approach in [24] is designed to handle articulations for pedestrian detection.

*Cow dataset* This dataset [5] is from the PASCAL Object Recognition Database Collection. There are 111 images in which cows appear at various positions. Since no training part is provided, we divided the dataset into two parts. We used first 55 images to train our detector, and tested it on the remaining 56 images. Then we trained on the second part, and tested on the first 55 images. This way we are able to report our performance on the whole dataset. The detection rates are shown in Table 2. Again we report a substantial increase in the detection rate by over 17% of NOCD in comparison to OCD. Interestingly, OCD is not able to improve the performance of pure chamfer distance. For this dataset, we used  $\lambda = 0.2$ , which indicates that the orientation information is not particularly useful. This is most likely due to a particular kind of background clutter present in this dataset as can be seen in the example result images in Fig. 6. The areas with dense vertical lines in the edge maps confused oriented chamfer matching. Oriented chamfer matching could not tell the ground truth location from such noise, since most of the false alarms appear in that area. The proposed NOCD was able to learn the difference between such noise and the true targets. For images with little clutter in the background, both OCD and NOCD performed equally well.

The performance of NOCD on this dataset also compares favorably to a very sophisticated learning and inference approach published very recently by Zhu et al. [19]. This comparison may not be quite fair, since this approach uses one-example learning, while our flexible cow model is constructed from 5 cow contours. However, on the other hand our detection algorithm is a simple max-sum. Thus, we do not employ any sophisticated inference in the detection process.

Chamfer distance	73.9%	proposed NOCD	91.0%
OCD	73.9%	Zhu et al. [19]	88.2%

**Table 2.** Detection rate on Cow Dataset.

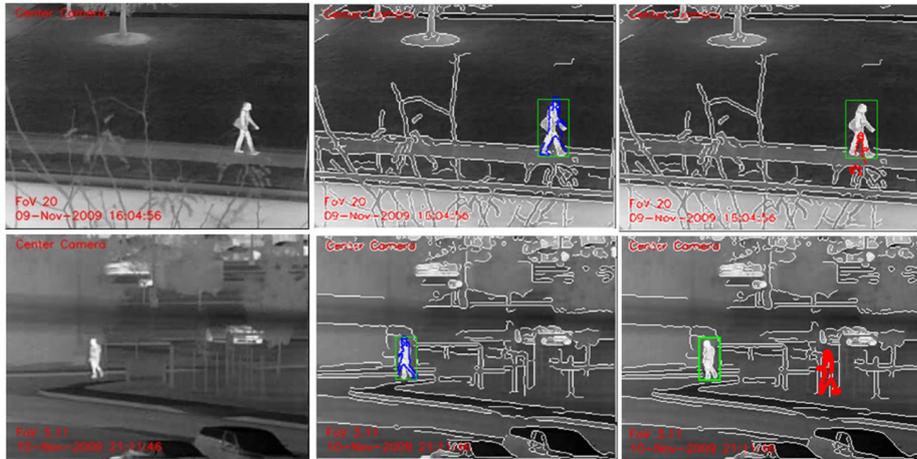


**Fig. 6.** Example detection results on the cow dataset. Left column NOCD. Right column OCD. Green rectangle denotes the ground truth object location.

*Infrared images* Without extra training, we use the same human model and the same normalizers as for TU Darmstadt Pedestrian dataset to carry out several tests on infrared images. In these images, humans are small, about  $60 \times 40$  pixels, which increase the possibility of misalignment to background. Some detection results are shown in Fig. 7.

## 9 Conclusions

By adding the term of orientation in the evaluation of the score, oriented chamfer distance is more robust to accidental alignment to the background noise than chamfer distance. However, as our experimental results clearly demonstrate this still does not solve the problem of matching to cluttered background, which often leads to a better score than the score at true object location. The proposed NOCD provides a solution to this problem by utilizing AdaBoost to learn normalization of OCD. The key idea is to compare the chamfer matching score of a given template to scores of a set of normalizers. The obtained ratios are interpreted as weak learners, and the strong learner obtained by AdaBoost is interpreted as a normalized OCD. Based on specific application, the proposed method could be modified by replacing oriented chamfer distance with oriented Hausdorff distance, or using sparse logistic regression instead of Adaboost in training phase.



**Fig. 7.** Detection result for infrared images. The original images are in the first column. The second column shows result of NOCD while the third column shows the results of OCD. Blue and red dots represent the corresponding parts of the model. Green rectangle denotes the ground truth bounding box. The edge map is overlaid in white on the original images.

## Acknowledgments

The work has been supported by the NSF Grants IIS-0812118, BCS-0924164, the AFOSR Grant FA9550-09-1-0207, and the DOE Award 71498-001-09.

## References

1. M. Andriluka, S. Roth, and B. Schiele. People-Tracking-by-Detection and People-Detection-by-Tracking. *CVPR*, 2008.
2. H. G. Barrow, J. M. Tenenbaum, R. C. Bolles, and H. C. Wolf. Parametric correspondence and chamfer matching: Two new techniques for image matching. *In Proc. 5th Int. Joint Conf. Artificial Intelligence*, pages 659-663, 1977.
3. D. M. Gavrilu and S. Munder. Multi-cue Pedestrian Detection and Tracking from a Moving Vehicle. *International Journal of Computer Vision*, 73(1), 41-9, 2007.
4. B. Leibe, E. Seemann, and B. Schiele. Pedestrian Detection in Crowded Scenes. *IEEE Conf. on Computer Vision and Pattern Recognition*, 2005.
5. Bastian Leibe, Ales Leonardis, and Bernt Schiele. Combined object categorization and segmentation with an implicit shape model. *ECCV'04 Workshop on Statistical Learning in Computer Vision*, 2004.
6. B. Stenger, A. Thayananthan, P. H. S. Torr, and R. Cipolla. Model-based hand tracking using a hierarchical bayesian filter. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 28(9), pages 1372-1384, 2006
7. A. Opelt, A. Pinz, and A. Zisserman. A Boundary-Fragment-Model for Object Detection. *Lecture Notes in Computer Science*, 3952, pages 575-588, 2006

8. A. Opelt, A. Pinz, and A. Zisserman. Incremental learning of object detectors using a visual shape alphabet. *In Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Volume 1, pages 3-10, 2006
9. G. Heitz, G. Elidan, B. Packer, and D. Koller. Shape-Based Object Localization for Descriptive Classification. *International Journal of Computer Vision*, Volume 84, No.1, August, 2009
10. M. Van Herk. Image Registration Using Chamfer Matching. *Handbook of medical imaging: processing and analysis*, Academic Press, 2000
11. O. Nomira and M. Abdel-Mottaleb. Hierarchical contour matching for dental X-ray radiographs. *Pattern Recognition*, 41(1), pages 130-138, 2008
12. S. Faisan, N. Passat, V. Noblet, R. Chabrier, and C. Meyer. Topology Preserving Warping of Binary Images: Application to Atlas-Based Skull Segmentation. *Medical Image Computing and Computer-Assisted Intervention*, pages 211-218, 2008
13. B. Wu and R. Nevatia. Detection and Segmentation of Multiple, Partially Occluded Objects by Grouping, Merging, Assigning Part Detection Responses. *International Journal of Computer Vision*, 82(2) pages 185, 2009
14. A. Thayananthan, B. Stenger, P. Torr, and R. Cipolla. Shape context and chamfer matching in cluttered scenes. *In Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pages 127-133, 2003.
15. S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 24(24), pages, 509-522, 2002.
16. G. Borgefors. Hierarchical chamfer matching: A parametric edge matching algorithm. *IEEE Trans. Pattern Analysis and Machine Intell.*, 10(6):849-865, 1988.
17. J. Shotton, A. Blake, and R. Cipolla. Multi-scale categorical object recognition using contour fragments. *IEEE Trans. Pattern Analysis and Machine Intell.*, 30(7):1270-1281, 2008.
18. Y. Freund and R. Schapire. A decision theoretic generalisation of online learning. *Computer and System Sciences*, 55(1):119-139, 1997.
19. L. Zhu, Y. Chen, and A. Yuille. Learning a Hierarchical Deformable Template for Rapid Deformable Object Parsing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 99(1), 2009.
20. Y. Lee and K. Grauman. Shape Discovery from Unlabeled Image Collections. *IEEE Conf. on Computer Vision and Pattern Recognition*, 2254 - 2261, Miami, FL, USA, June, 2009.
21. A. Torralba, K. Murphy, and W. Freeman. Sharing Features: Efficient Boosting Procedures for Multiclass Object Detection. *IEEE Conf. on Computer Vision and Pattern Recognition*, pages, 762-769, 2004.
22. P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. *IEEE Conf. on Computer Vision and Pattern Recognition*, 2001.
23. N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. *IEEE Conf. on Computer Vision and Pattern Recognition*, 2005.
24. E. Seemann and B. Schiele. Cross-articulation learning for robust detection of pedestrians. *DAGM*, 2006.
25. NH. Trinh and BB. Kimia. Category-Specific Object Recognition and Segmentation Using a Skeletal Shape Model. *Proceedings of the Twentieth British Machine Vision Conference*, London, UK. September, 2009.
26. D. P. Huttenlocher, G. A. Klanderma, and W. J. Rucklidge. Comparing Images Using the Hausdorff Distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(9): 850-863, 1993.
27. C. F. Olson and D.P. Huttenlocher. Automatic Target Recognition by Matching Oriented Edge Pixels. *IEEE Transactions on Image Processing*, 6(1): 103-113, 1997.