Conservative Wasserstein Training for Pose Estimation

Xiaofeng Liu[1,2†*], Yang Zou[1†], Tong Che[3†], Peng Ding[4], Ping Jia[4], Jane You[5], B.V.K. Vijaya Kumar[1]

[1]Carnegie Mellon University; [2]Harvard University; [3]MILA
[4]CIOMP, Chinese Academy of Sciences; [5]The Hong Kong Polytechnic University

[†]Contribute equally [*]Corresponding to: liuxiaofengcmu@gmail.com

Abstract

This paper targets the task with discrete and periodic class labels (*e.g.*, pose/orientation estimation) in the context of deep learning. The commonly used cross-entropy or regression loss is not well matched to this problem as they ignore the periodic nature of the labels and the class similarity, or assume labels are continuous value. We propose to incorporate inter-class correlations in a Wasserstein training framework by pre-defining (*i.e.*, using arc length of a circle) or adaptively learning the ground metric. We extend the ground metric as a linear, convex or concave increasing function *w.r.t.* arc length from an optimization perspective. We also propose to construct the conservative target labels which model the inlier and outlier noises using a wrapped unimodal-uniform mixture distribution. Unlike the one-hot setting, the conservative label makes the computation of Wasserstein distance more challenging. We systematically conclude the practical closed-form solution of Wasserstein distance for pose data with either one-hot or conservative target label. We evaluate our method on head, body, vehicle and 3D object pose benchmarks with exhaustive ablation studies. The Wasserstein loss obtaining superior performance over the current methods, especially using convex mapping function for ground metric, conservative label, and closed-form solution.

1. Introduction

There are some prediction tasks where the output labels are discrete and are periodic. For example, consider the problem of pose estimation. Although pose can be a continuous variable, in practice, it is often discretized *e.g.*, in 5-degree intervals. Because of the periodic nature of pose, a 355-degree label is closer to 0-degree label than the 10-degree label. Thus it is important to consider the periodic and discrete nature of the pose classification problem.

In previous literature, pose estimation is often cast as a multi-class classification problem [?], a metric regression problem [?], or a mixture of both [?].

In a multi-class classification formulation using the cross-entropy (CE) loss, the class labels are assumed to be independent from each other [?, ?, ?]. Therefore, the inter-class similarity is not properly exploited. For instance, in Fig. 1, we would prefer the predicted probability distribution to be concentrated near the ground truth class, while the CE loss does not encourage that.

On the other hand, metric regression methods treat the pose as a continuous numerical value [?, ?, ?], although the label of pose itself is discrete. As manifested in [?, ?, ?], learning a regression model using discrete labels will cause over-fitting and exhibit similar or inferior performance compared with classification.

Recent works either use a joint classification and regression loss [?] or divide a circle into several sectors with a coarse classification that ignores the periodicity, and then applying regression networks to each sector independently as an ordinal regression problem [?]. Unfortunately, none of them fundamentally address the limitations of CE or regression loss in angular data.

In this work, we employ the Wasserstein loss as an alternative for empirical risk minimization. The 1^{st} Wasserstein distance is defined as the cost of optimal transport for moving the mass in one distribution to match the target distribution [?, ?]. Specifically, we measure the Wasserstein distance between a softmax prediction and its target label, both of which are normalized as histograms. By defining the ground metric as class similarity, we can measure prediction performance in a way that is sensitive to correlations between the classes.

The ground metric can be predefined when the similarity structure is known a priori to incorporate the inter-class correlation, *e.g.*, the arc length for the pose. We further extend the arc length to its increasing function from an optimization perspective. The exact

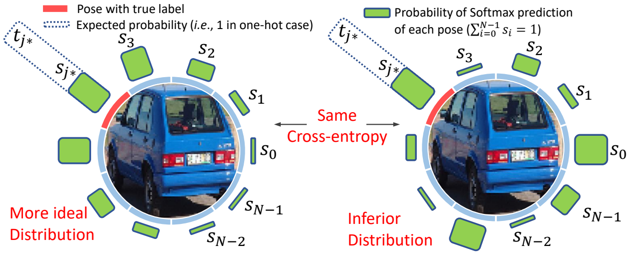


Figure 1. The limitation of CE loss for pose estimation. The ground truth direction of the car is t_j*. Two possible softmax predictions (green bar) of the pose estimator have the same probability at t_j* position. Therefore, both predicted distributions have the same CE loss. However, the left prediction is preferable to the right, since we desire the predicted probability distribution to be larger and closer to the ground truth class.

Wasserstein distance in one-hot target label setting can be formulated as a soft-attention scheme of all prediction probabilities and be rapidly computed. We also propose to learn the optimal ground metric following alternative optimization.

Another challenge of pose estimation comes from low image quality (*e.g.*, blurry, low resolution) and the consequent noisy labels. This requires 1) modeling the noise for robust training [?, ?] and 2) quantifying the uncertainty of predictions in testing phase [?].

Wrongly annotated targets may bias the training process [?, ?]. We instigate two types of noise. The outlier noise corresponds to one training sample being very distant from others by random error and can be modeled by a uniform distribution [?]. We notice that the pose data is more likely to have inlier noise where the labels are wrongly annotated as the near angles and propose to model it using a unimodal distribution. Our solution is to construct a conservative target distribution by smoothing the one-hot label using a wrapped uniform-unimodal mixture model.

Unlike the one-hot setting, the conservative target distribution makes the computation of Wasserstein distance more advanced because of the numerous possible transportation plans. The $\mathcal{O}(N^3)$ computational complexity for N classes has long been a stumbling block in using Wasserstein distance for large-scale applications. Instead of only obtaining its approximate solution using a $\mathcal{O}(N^2)$ complexity algorithm [?], we systematically analyze the fast closed-form computation of Wasserstein distance for our conservative label when our ground metric is a linear, convex, or concave increasing function *w.r.t.* the arc length. The linear and convex cases can be solved with linear complexity of $\mathcal{O}(N)$. Our *exact* solutions are significantly more efficient than the approximate baseline.

The main contributions of this paper are summarized as

- We cast the pose estimation as a Wasserstein training problem. The inter-class relationship of angular data is explicitly incorporated as prior information in our ground metric which can be either pre-defined (a function *w.r.t.* arc length) or adaptively learned with alternative optimization.
- We model the inlier and outlier error of pose data using a wrapped discrete unimodal-uniform mixture distribution, and regularize the target confidence by transforming one-hot label to conservative target label.
- For either one-hot or conservative target label, we systematically conclude the possible fast closed-form solution when a non-negative linear, convex or concave increasing mapping function is applied in ground metric.

We empirically validate the effectiveness and generality of the proposed method on multiple challenging benchmarks and achieve the state-of-the-art performance.

2. Related Works

Pose or viewpoint estimation has a long history in computer vision [?]. It arises in different applications, such as head [?], pedestrian body [?], vehicle [?] and object class [?] orientation/pose estimation. Although these systems are mostly developed independently, they are essentially the same problem in our framework.

The current related literature using deep networks can be divided into two categories. Methods in the first group, such as [?, ?, ?], predict keypoints in images and then recover the pose using pre-defined 3D object models. The keypoints can be either semantic [?, ?, ?] or the eight corners of a 3D bounding box encapsulating the object [?, ?].

The second category of methods, which are more close to our approach, estimate angular values directly from the image [?, ?]. Instead of the typical Euler angle representation for rotations [?], biternion representation is chosen in [?, ?] and inherits the periodicity in its *sin* and *cos* operations. However, their setting is compatible with only the regression. Several studies have evaluated the performance of classification and regression-based loss functions and conclude that the classification methods usually outperform the regression ones in pose estimation [?, ?].

These limitations were also found in the recent approaches which combine classification with regression or even triplet loss [?, ?].

Wasserstein distance is a measure defined between probability distributions on a given metric space [?]. Recently, it attracted much attention in generative models *etc* [?]. [?] introduces it for multi-class multi-label task with a linear model. Because of the significant amount of computing needed to solve the exact distance for general cases, these methods choose the approximate solution, whose complexity is still in $\mathcal{O}(N^2)$ [?]. The fast computing of discrete Wasserstein distance is also closely related to SIFT [?] descriptor, hue in HSV or LCH space [?] and sequence data [?]. Inspired by the above works, we further adapted this idea to the pose estimation, and encode the geometry of label space by means of the ground matrix. We show that the fast algorithms exist in our pose label structure using the one-hot or conservative target label and the ground metric is not limited to the arc length.

Robust training with noise data has long been studied for general classification problems [?]. Smoothing the one-hot label [?] with a uniform distribution or regularizing the entropy of softmax output [?] are two popular solutions. Some works of regression-based localization model the uncertainty of point position in a plane with a 2D Gaussian distribution [?]. [?] propose to regularize self-training with confidence. However, there are few studies for the discrete periodic label. Besides sampling on Gaussian, the Poisson and the Binomial distribution are further discussed to form a unimodal-uniform distribution.

Uncertainty quantification of pose estimation aims to quantify the reliability of a result *e.g.*, a confidence distribution of each class rather than a certain angle value for pose data [?]. A well-calibrated uncertainty is especially important for large systems to assess the consequence of a decision [?, ?]. [?] proposes to output numerous sets of the mean and variation of Gaussian/Von-Mises distribution following [?]. It is unnecessarily complicated and is a somewhat ill-matched formulation as it assumes the pose label is continuous, while it is discrete. We argue that the $softmax$ is a natural function to capture discrete uncertainty, and is compatible with Wasserstein training.

3. Methodology

We consider learning a pose estimator h_θ, parameterized by θ, with N-dimensional softmax output unit. It maps a image x to a vector $\mathrm{s} \in \mathbb{R}^N$. We perform learning over a hypothesis space $\mathcal{H}$ of h_θ. Given input x and its target ground truth one-hot label t, typically, learning is performed via empirical risk minimization to solve $\min_{h_\theta \in \mathcal{H}} \mathcal{L}(h_\theta(\mathrm{x}), \mathrm{t})$, with a loss $\mathcal{L}(\cdot,\cdot)$ acting as a surrogate of performance measure.

Unfortunately, cross-entropy, information divergence, Hellinger distance and $\mathcal{X}^2$ distance-based loss treat the output dimensions independently [?], ignoring the similarity structure on pose label space.

Let $\mathrm{s} = \{s_i\}_{i=0}^{N-1}$ be the output of $h_\theta(\mathrm{x})$, *i.e.*, softmax prediction with N classes (angles), and define $\mathrm{t} = \{t_j\}_{j=0}^{N-1}$ as the target label distribution, where $i, j \in \{0, \cdots, N-1\}$ be the index of dimension (class). Assume class label possesses a ground metric $\mathrm{D}_{i,j}$, which measures the semantic similarity between i-th and j-th dimensions of the output. There are N^2 possible $\mathrm{D}_{i,j}$ in a N class dataset and form a ground distance matrix $\mathrm{D} \in \mathbb{R}^{N\times N}$. When s and t are both histograms, the discrete measure of exact Wasserstein loss is defined as

$$\mathcal{L}_{\mathrm{D}_{i,j}}(\mathrm{s},\mathrm{t}) = \inf_{\mathrm{W}} \sum_{j=0}^{N-1} \sum_{i=0}^{N-1} \mathrm{D}_{i,j}\mathrm{W}_{i,j} \tag{1}$$

where W is the transportation matrix with $\mathrm{W}_{i,j}$ indicating the mass moved from the i^{th} point in source distribution to the j^{th} target position. A valid transportation matrix W satisfies: $\mathrm{W}_{i,j} \geq 0$; $\sum_{j=0}^{N-1} \mathrm{W}_{i,j} \leq s_i$; $\sum_{i=0}^{N-1} \mathrm{W}_{i,j} \leq t_j$; $\sum_{j=0}^{N-1}\sum_{i=0}^{N-1} \mathrm{W}_{i,j} = \min(\sum_{i=0}^{N-1} s_i, \sum_{j=0}^{N-1} t_j)$.

The ground distance matrix D in Wasserstein distance is usually unknown, but it has clear meanings in our application. Its i, j-th entry $\mathrm{D}_{i,j}$ could be the geometrical distance between the i-th and j-th points in a circle. A possible choice is using the arc length $d_{i,j}$ of a circle (*i.e.*, ℓ_1 distance between the i-th and j-th points in a circle) as the ground metric $\mathrm{D}_{i,j} = d_{i,j}$.

$$d_{i,j} = \min\{|i-j|, N-|i-j|\} \tag{2}$$

The Wasserstein distance is identical to the Earth mover's distance when the two distributions have the same total masses (*i.e.*, $\sum_{i=0}^{N-1} s_i = \sum_{j=0}^{N-1} t_j$) and using the symmetric distance $d_{i,j}$ as $\mathrm{D}_{i,j}$.

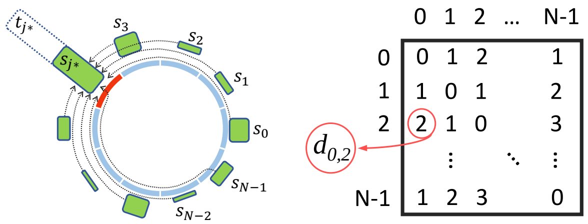
This setting is satisfactory for comparing the similarity of SIFT or hue [?], which do not use a neural network optimization. The previous efficient algorithm usually holds only for $\mathrm{D}_{i,j} = d_{i,j}$. We propose to extend the ground metric in $\mathrm{D}_{i,j}$ as $f(d_{i,j})$, where f is a positive increasing function *w.r.t.* $d_{i,j}$.

3.1. Wasserstein training with one-hot target

The one-hot encoding is a typical setting for multi-class one-label dataset. The distribution of a target label probability is $\mathrm{t} = \delta_{j,j^*}$, where j^* is the ground truth class, δ_{j,j^*} is a Dirac delta, which equals to 1 for $j = j^*$[1], and 0 otherwise.

Theorem 1. Assume that $\sum_{j=0}^{N-1} t_j = \sum_{i=0}^{N-1} s_i$, and t

[1] We use i, j interlaced for s and t, since they index the same group of positions in a circle.



$$\begin{array}{c|cccccc}
 & 0 & 1 & 2 & \dots & & N-1 \\
\hline
0 & 0 & 1 & 2 & & & 1 \\
1 & 1 & 0 & 1 & & & 2 \\
2 & 2 & 1 & 0 & & & 3 \\
 & & \vdots & & & \ddots & \vdots \\
N-1 & 1 & 2 & 3 & & & 0
\end{array}$$

Figure 2. Left: The only possible transport plan in one-hot target case. Right: the ground matrix using arc length as ground metric.

is a one-hot distribution with $t_{j^*} = 1$(or $\sum_{i=0}^{N-1} s_i$)[2], there is only one feasible optimal transport plan.

According to the criteria of W, all masses have to be transferred to the cluster of the ground truth label j^*, as illustrated in Fig. 2. Then, the Wasserstein distance between softmax prediction s and one-hot target t degenerates to

$$\mathcal{L}_{\mathrm{D}^f_{i,j}}(\mathrm{s},\mathrm{t}) = \sum_{i=0}^{N-1} s_i f(d_{i,j^*}) \qquad (3)$$

where $\mathrm{D}^f_{i,j} = f(d_{i,j})$. Practically, f can be an increasing function proper, *e.g.*, p^{th} power of $d_{i,j}$ and Huber function. The exact solution of Eq. (3) can be computed with a complexity of $\mathcal{O}(N)$. The ground metric term $f(d_{i,j^*})$ works as the weights *w.r.t.* s_i, which takes all classes into account following a soft attention scheme [?, ?, ?]. It explicitly encourages the probabilities distributing on the neighboring classes of j^*. Since each s_i is a function of the network parameters, differentiating $\mathcal{L}_{\mathrm{D}^f_{i,j}}$ *w.r.t.* network parameters yields $\sum_{i=0}^{N-1} s_i' f(d_{i,j^*})$.

In contrast, the cross-entropy loss in one-hot setting can be formulated as $-1\log s_{j^*}$, which only considers a single class prediction like the hard attention scheme [?, ?, ?], that usually loses too much information. Similarly, the regression loss using softmax prediction could be $f(d_{i^*,j^*})$, where i^* is the class with maximum prediction probability.

In addition to the predefined ground metric, we also propose to learn D adaptively along with our training following an alternative optimization scheme [?].
Step 1: Fixing ground matrix D to compute $\mathcal{L}_{\mathrm{D}_{i,j}}(\mathrm{s},\mathrm{t})$ and updating the network parameters.
Step 2: Fixing network parameters and postprocessing D using the feature-level ℓ_1 distances between different poses.

We use the normalized second-to-last layer neural response in this round as feature vector, since there is no subsequent non-linearities. Therefore, it is meaningful to average the feature vectors in each pose class to compute their centroid and reconstruct $\mathrm{D}_{i,j}$ using the ℓ_1 distances between these centroids $\overline{d}_{i,j}$. To avoid the model collapse, we construct the $\mathrm{D}_{i,j} = \frac{1}{1+\alpha}\left\{f(\overline{d}_{i,j}) + \alpha f(d_{i,j})\right\}$ in each round, and decrease α from 10 to 0 gradually in the training.

3.2. Wrapped unimodal-uniform smoothing

The outlier noise exists in most of data-driven tasks, and can be modeled by a uniform distribution [?]. However, pose labels are more likely to be mislabeled as a close class of the true class. It is more reasonable to construct a unimodal distribution to depict the inlier noise in pose estimation, which has a peak at class j^* while decreasing its value for farther classes. We can sample on a continuous unimodal distribution (*e.g.*, Gaussian distribution) and followed by normalization, or choose a discrete unimodal distribution (*e.g.*, Poisson/Binomial distribution).
Gaussian/Von-Mises Distribution has the probability density function (PDF) $f(x) = \frac{\exp\left\{-(x-\mu)^2/2\sigma^2\right\}}{\sqrt{2\pi\sigma^2}}$ for $x \in [0, K]$, where $\mu = K/2$ is the mean, and σ^2 is the variance. Similarly, the Von-Mises distribution is a close approximation to the circular analogue of the normal distribution (*i.e.*, $K = N-1$). We note that the geometric loss [?] is a special case, when we set $\xi = 1, \eta = 0$, $K = N-1$, remove the normalization and adopt CE loss. Since we are interested in modeling a discrete distribution for target labels, we simply apply a softmax operation over their PDF. Note that the output values are mapped to be defined on the circle.
Poisson Distribution is used to model the probability of the number of events, k occurring in a particular interval of time. Its probability mass function (PMF) is:

$$p_k = \frac{\lambda^k \exp(-\lambda)}{k!}, \quad k = 0, 1, 2, ..., \qquad (4)$$

where $\lambda \in \mathbb{R}^+$ is the average frequency of these events. We can sample $K+1$ probabilities (*i.e.*, $0 \le k \le K$) on this PMF and followed by normalization for discrete unimodal probability distributions. Since its mean and variation are the same (*i.e.*, λ), it maybe inflexible to adjust its shape.
Binomial Distribution is commonly adopted to model the probability of a given number of successes out of a given number of trails k and the success probability p.

$$p_k = \binom{n}{k} p^k (1-p)^{n-k}, \quad n \in \mathbb{N}, \quad k = 0, 1, 2, ..., n \quad (5)$$

We set $n = K$ to construct a distribution with $K+1$

[2] We note that softmax cannot strictly guarantee the sum of its outputs to be 1 considering the rounding operation. However, the difference of setting t_{j^*} to 1 or $\sum_{i=0}^{N-1} s_i$) is not significant in our experiments using the typical format of softmax output which is accurate to 8 decimal places.

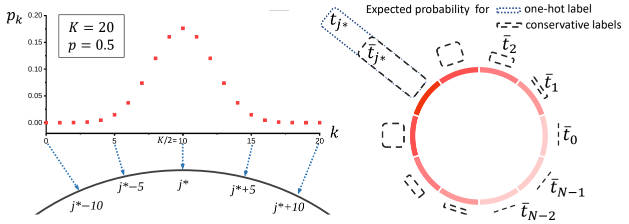


Figure 3. Left: the wrapping operation with a Binomial distribution ($K+1$ is the number of involved classes of unimodal distribution). Right: the distribution of conservative target label.

bins without softmax normalization. Its warp processing with $K = 20$ is illustrated in Fig. 3.

The conservative target distribution $\bar{\mathrm{t}}$ is constructed by replacing t_j in t with $(1-\xi-\eta)t_j+\xi p_j+\eta\frac{1}{N}$, which can be regarded as the weighted sum of the original label distribution t and a unimodal-uniform mixture distribution. When we only consider the uniform distribution and utilize the CE loss, it is equivalent to label smoothing [?], a typical mechanism for outlier noisy label training, which encourages the model to accommodate less-confident labels.

By enforcing s to form a unimodal-uniform mixture distribution, we also implicitly encourage the probabilities to distribute on the neighbor classes of j^*.

3.3. Wasserstein training with conservative target

With the conservative target label, the fast computation of Wasserstein distance in Eq. (3) does not apply. A straightforward solution is to regard it as a general case and solve its closed-form result with a complexity higher than $\mathcal{O}(N^3)$ or get an approximate result with a complexity in $\mathcal{O}(N^2)$. The main results of this section are a series of analytic formulation when the ground metric is a nonnegative increasing linear/convex/concave function *w.r.t.* arc length with a reasonable complexity.

3.3.1 Arc length $d_{i,j}$ as the ground metric.

When we use $d_{i,j}$ as ground metric directly, the Wasserstein loss $\mathcal{L}_{d_{i,j}}(\mathrm{s},\bar{\mathrm{t}})$ can be written as

$$\mathcal{L}_{d_{i,j}}(\mathrm{s},\bar{\mathrm{t}}) = \inf_{\alpha\in\mathbb{R}} \sum_{j=0}^{N-1} |\sum_{i=0}^{j}(s_i-\bar{t}_i)-\alpha| \tag{6}$$

To the best of our knowledge, Eq. (6) was first developed in [?], in which it is proved for sets of points with unitary masses on the circle. A similar conclusion for the Kantorovich-Rubinstein problem was derived in [?, ?], which is known to be identical to the Wasserstein distance problem when $\mathrm{D}_{i,j}$ is a distance. We note that this is true for $\mathcal{L}_{d_{i,j}}$ (but false for $\mathcal{L}_{\mathrm{D}^{\rho}}(\mathrm{s},\bar{\mathrm{t}})$ with $\rho > 1$). The optimal α should be the median of the set values $\left\{\sum_{i=0}^{j}(s_i-\bar{t}_i), 0\leq j\leq N-1\right\}$ [?]. An equivalent distance is proposed from the circular cumulative distribution perspective [?]. All of these papers notice that computing Eq. (6) can be done in linear time (*i.e.*, $\mathcal{O}(N)$) weighted median algorithm (see [?] for a review).

We note that the partial derivative of Eq. (6) *w.r.t.* s_n is $\sum_{j=0}^{N-1}\mathrm{sgn}(\varphi_j)\sum_{i=0}^{j}(\delta_{i,n}-s_i)$, where $\varphi_j = \sum_{i=0}^{j}(s_i-\bar{t}_i)$, and $\delta_{i,n} = 1$ when $i = n$. Additional details are given in Appendix B.

3.3.2 Convex function *w.r.t.* $d_{i,j}$ as the ground metric

Next, we extend the ground metric as an nonnegative increasing and convex function of $d_{i,j}$, and show its analytic formulation. If we compute the probability with a precision ϵ, we will have $M = 1/\epsilon$ unitary masses in each distribution. We define the cumulative distribution function of s and $\bar{\mathrm{t}}$ and their pseudo-inverses as follows

$$\begin{array}{l} \mathrm{S}(i) = \sum_{i=0}^{N-1} s_i;\ \mathrm{S}(m)^{-1} = \inf\left\{i;\mathrm{S}(i)\geq m\right\} \\ \overline{\mathrm{T}}(i) = \sum_{i=0}^{N-1} \bar{t}_i;\ \overline{\mathrm{T}}(m)^{-1} = \inf\left\{i;\overline{\mathrm{T}}(i)\geq m\right\} \end{array} \tag{7}$$

where $m\in\left\{\frac{1}{M},\frac{2}{M},\cdots,1\right\}$. Following the convention $\mathrm{S}(i+N) = \mathrm{S}(i)$, S can be extended to the whole real number, which consider S as a periodic (or modulo [?]) distribution on $\mathbb{R}$.

Theorem 2. Assuming the arc length distance $d_{i,j}$ is given by Eq. (2) and the ground metric $\mathrm{D}_{i,j} = f(d_{i,j})$, with f a nonnegative, increasing and convex function. Then

$$\mathcal{L}_{\mathrm{D}_{i,j}^{conv}}(\mathrm{s},\bar{\mathrm{t}}) = \inf_{\alpha\in\mathbb{R}} \sum_{m=\frac{1}{M}}^{1} f(|\mathrm{S}(m)^{-1}-(\overline{\mathrm{T}}(m)-\alpha)^{-1}|) \tag{8}$$

where α is a to-be-searched transportation constant. A proof of Eq. (8) *w.r.t.* the continuous distribution was given in [?], which shows it holds for any couple of desecrate probability distributions. Although that proof involves some complex notions of measure theory, that is not needed in the discrete setting. The proof is based on the idea that the circle can always be “cut” somewhere by searching for a m, that allowing us to reduce the modulo problem [?] to ordinal case. Therefore, Eq. (8) is a generalization of the ordinal data. Actually, we can also extend Wasserstein distance for discrete distribution in a line [?] as

$$\sum_{m=\frac{1}{M}}^{1} f(|\mathrm{S}(m)^{-1}-\overline{\mathrm{T}}(m)^{-1}|) \tag{9}$$

where f can be a nonnegative linear/convex/concave increasing function *w.r.t.* the distance in a line. Eq. (9) can be computed with a complexity of $\mathcal{O}(N)$ for two discrete distributions. When f is a convex function, the optimal α can be found with a complexity of $\mathcal{O}(\log M)$ using the Monge condition[3] (similar to binary search). Therefore, the exact solution of Eq. (8) can be obtained with $\mathcal{O}(N\log M)$ complexity. In practice, $\log M$ is a constant ($\log 10^8$) according to the precision of softmax predictions, which is much smaller than N (usually $N = 360$ for pose data).

Here, we give some measures[4] using the typical convex ground metric function.

$\mathcal{L}_{\mathrm{D}^{\rho}_{i,j}}(\mathrm{s}, \bar{\mathrm{t}})$, the Wasserstein measure using d^{ρ} as ground metric with $\rho = 2, 3, \cdots$. The case $\rho = 2$ is equivalent to the Cramér distance [?]. Note that the Cramér distance is not a distance metric proper. However, its square root is.

$$\mathrm{D}^{\rho}_{i,j} = d^{\rho}_{i,j} \tag{10}$$

$\mathcal{L}_{\mathrm{D}^{H\tau}_{i,j}}(\mathrm{s}, \bar{\mathrm{t}})$, the Wasserstein measure using a Huber cost function with a parameter τ.

$$\mathrm{D}^{H\tau}_{i,j} = \begin{cases} d^2_{i,j} & \text{if } d_{i,j} \leq \tau \\ \tau(2d_{i,j} - \tau) & \text{otherwise.} \end{cases} \tag{11}$$

3.3.3 Concave function *w.r.t.* $d_{i,j}$ as the ground metric

In practice, it may be useful to choose the ground metric as a nonnegative, concave and increasing function *w.r.t.* $d_{i,j}$. For instance, we can use the chord length.

$$\mathrm{D}^{chord}_{i,j} = 2r\,\sin(d_{i,j}/2r) \tag{12}$$

where $r = N/2\pi$ is the radius. Therefore, $f(\cdot)$ can be regarded as a concave and increasing function on interval $[0,N/2]$ *w.r.t.* $d_{i,j}$.

It is easy to show that $\mathrm{D}^{chord}_{i,j}$ is a distance, and thus $\mathcal{L}_{\mathrm{D}^{chord}}(\mathrm{s}, \bar{\mathrm{t}})$ is also a distance between two probability distributions [?]. Notice that a property of concave distance is that they do not move the mass shared by the s and $\bar{\mathrm{t}}$ [?]. Considering the Monge condition does not apply for concave function, there is no corresponding fast algorithm to compute its closed-form solution. In most cases, we settle for linear programming. However, the simplex or interior point algorithm are known to have at best a $\mathcal{O}(N^{2.5}\log(ND_{max}))$ complexity to compare two histograms on N bins [?, ?], where $D_{max} = f(\frac{N}{2})$ is the maximal distance between the two bins.

Method	MAAD
BIT[?]	25.2°
DDS[?]†	23.7°
$\mathcal{L}_{d_{i,j}}(\mathrm{s},\mathrm{t})$	18.8°
$\mathcal{L}_{\mathrm{D}^2_{i,j}}(\mathrm{s},\mathrm{t})$	17.1°
$\mathcal{L}_{\mathrm{D}^{chord}_{i,j}}(\mathrm{s},\mathrm{t})$	19.1°

Method	MAAD
A-$\mathcal{L}_{d_{i,j}}(\mathrm{s},\mathrm{t})$	17.5°
A-$\mathcal{L}_{\mathrm{D}^2_{i,j}}(\mathrm{s},\mathrm{t})$	$\underline{17.3}$°
$\approx \mathcal{L}_{d_{i,j}}(\mathrm{s},\mathrm{t})$	19.0°
$\approx \mathcal{L}_{\mathrm{D}^2_{i,j}}(\mathrm{s},\mathrm{t})$	17.8°
$\approx \mathcal{L}_{\mathrm{D}^{chord}_{i,j}}(\mathrm{s},\mathrm{t})$	19.5°

Table 1. Results on CAVIAR head pose dataset (the lower MAAD the better).† Our implementation based on their publicly available codes. The best are in bold while the second best are underlined.

Although the general computation speed of the concave function is not satisfactory, the step function $f(t) = \mathbb{1}_{t\neq 0}$ (one every where except at 0) can be a special case, which has significantly less complexity [?]. Assuming that the $f(t) = \mathbb{1}_{t\neq 0}$, the Wasserstein metric between two normalized discrete histograms on N bins is simplified to the ℓ_1 distance.

$$\mathcal{L}_{\mathbb{1}_{d_{i,j}\neq 0}}(\mathrm{s}, \bar{\mathrm{t}}) = \frac{1}{2}\sum_{i=0}^{N-1}|\mathrm{s}_i - \bar{\mathrm{t}}_i| = \frac{1}{2}||\mathrm{s} - \bar{\mathrm{t}}||_1 \tag{13}$$

where $||\cdot||_1$ is the discrete ℓ_1 norm.

Unfortunately, its fast computation is at the cost of losing the ability to discriminate the difference of probability in a different position of bins.

4. Experiments

In this section, we show the implementation details and experimental results on the head, pedestrian body, vehicle and 3D object pose/orientation estimation tasks. To illustrate the effectiveness of each setting choice and their combinations, we give a series of elaborate ablation studies along with the standard measures.

We use the prefix A and $\approx$ denote the adaptively ground metric learning (in Sec. 3.1) and approximate computation of Wasserstein distance [?, ?] respectively. (s, t) and $(\mathrm{s}, \bar{\mathrm{t}})$ refer to using one-hot or conservative target label. For instance, $\mathcal{L}_{d_{i,j}}(\mathrm{s}, \mathrm{t})$ means choosing Wasserstein loss with arc length as ground metric and using one-hot target label.

4.1. Head pose

Following [?, ?], we choose the occluded version of CAVIAR dataset [?] and construct the training, validation and testing using 10802, 5444 and 5445 identity-independent images respectively. Since the orientation of gaze is coarsely labeled, and almost 40% training samples lie within ±4° of the four canonical orientations, regression-based methods [?, ?] are inefficient.

For fair comparison, we use the same deep batch normalized VGG-style [?] backbone as in [?, ?]. Instead

[3] $\mathrm{D}_{i,j}+\mathrm{D}_{i',j'} < \mathrm{D}_{i,j'}+\mathrm{D}_{i',j}$ whenever $i < i'$ and $j < j'$.

[4] We refer to "measure", since a ρ^{th}-root normalization is required to get a distance [?], which satisfies three properties: positive definiteness, symmetry and triangle inequality.

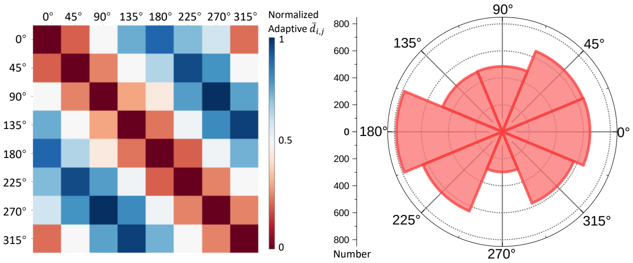


Figure 4. Normalized adaptively learned ground matrix and polar histogram *w.r.t.* the number of training samples in TUD dataset.

of a sigmoid unit in their regression model, the last layer is set to a softmax layer with 8 ways for Right, Right-Back, Back, Left-Back, Left, Left-Front, Front and Right-Front poses.

The metric used here is the mean absolute angular deviation (MAAD), which is widely adopted for angular regression tasks. The results are summarized in Table 1. The Wasserstein training boosts the performance significantly. Using convex f can further improve the result, while the losses with concave f are usually inferior to the vanilla Wasserstein loss with arc length as the ground metric. The adaptive ground metric learning is helpful for the $\mathcal{L}_{d_{i,j}}(\mathrm{s},\mathrm{t})$, but not necessary when we extend the ground metric to the square of $d_{i,j}$.

We also note that the exact Wasserstein distances are consistently better than their approximate counterparts [?]. More appealingly, in the training stage, $\mathcal{L}_{d_{i,j}}(\mathrm{s},\mathrm{t})$ is 5× faster than $\approx \mathcal{L}_{d_{i,j}}(\mathrm{s},\mathrm{t})$ and 3× faster than conventional regression-based method [?] to achieve the convergence.

4.2. Pedestrians orientation

The TUD multi-view pedestrians dataset [?] consists of 5,228 images along with bounding boxes. Its original annotations are relatively coarse, with only eight classes. We adopt the network in [?] and show the results in Table 2. Our methods, especially the $\mathcal{L}_{\mathrm{D}^2_{i,j}}(\mathrm{s},\bar{\mathrm{t}})$ outperform the cross-entropy loss-based approach in all of the eight classes by a large margin. The improvements in the case of binomial-uniform regularization ($\xi = 0.1, \eta = 0.05, K = 4, p = 0.5$) seems limited for 8 class setting, because each pose label covers 45° resulting in relatively low noise level.

The adaptive ground metric learning can contribute to higher accuracy than the plain $\mathcal{L}_{d_{i,j}}(\mathrm{s},\mathrm{t})$. Fig. 4 provides a visualization of the adaptively learned ground matrix. The learned $\bar{d}_{i,j}$ is slightly larger than $d_{i,j}$ when limited training samples are available in the related classes, *e.g.*, $d_{225^\circ,180^\circ} < d_{225^\circ,270^\circ}$. A larger ground metric value may emphasize the class with fewer samples in the training.

Method	0°	45°	90°	135°	180°	225°	270°	315°
CE loss[?]	0.90	0.96	0.92	1.00	0.92	0.88	0.89	0.95
$\mathcal{L}_{d_{i,j}}(\mathrm{s},\mathrm{t})$	0.93	0.97	0.95	1.00	0.96	0.91	0.91	0.95
A-$\mathcal{L}_{d_{i,j}}(\mathrm{s},\mathrm{t})$	0.94	0.97	0.96	1.00	0.95	0.92	0.91	0.96
$\mathcal{L}_{\mathrm{D}^2_{i,j}}(\mathrm{s},\mathrm{t})$	0.95	0.97	0.96	1.00	0.96	0.92	0.91	0.96
CE loss$(\mathrm{s},\bar{\mathrm{t}})$	0.90	0.96	0.94	1.00	0.92	0.90	0.90	0.95
$\mathcal{L}_{\mathrm{D}^2_{i,j}}(\mathrm{s},\bar{\mathrm{t}})$	0.95	0.98	0.96	1.00	0.96	0.93	0.92	0.96

Table 2. Class-wise accuracy for TUD pedestrian orientation estimation with 8 pose setting (the higher the better).

Method	Mean AE	$Acc_{\frac{\pi}{8}}$	$Acc_{\frac{\pi}{4}}$
RTF[?]	34.7	0.686	0.780
SHIFT[?]	22.6	0.706	0.861
$\mathcal{L}_{d_{i,j}}(\mathrm{s},\mathrm{t})$	19.1	0.748	0.900
A-$\mathcal{L}_{d_{i,j}}(\mathrm{s},\mathrm{t})$	20.5	0.723	0.874
$\mathcal{L}_{\mathrm{D}^2_{i,j}}(\mathrm{s},\mathrm{t})$	18.5	0.756	0.905
$\mathcal{L}_{\mathrm{D}^2_{i,j}}$ | SHIFT $(\mathrm{s},\bar{\mathrm{t}})_G$	$\underline{16.4}$ | 20.1	$\underline{0.764}$ | 0.724	$\underline{0.909}$ | 0.874
$\mathcal{L}_{\mathrm{D}^2_{i,j}}$ | SHIFT $(\mathrm{s},\bar{\mathrm{t}})_P$	17.7 | 20.8	0.760 | 0.720	0.907 | 0.871
$\mathcal{L}_{\mathrm{D}^2_{i,j}}$ | SHIFT $(\mathrm{s},\bar{\mathrm{t}})_B$	16.3 | 20.1	0.766 | 0.723	0.910 | 0.875
Human [?]	9.1	0.907	0.993

Table 3. Results on TUD pedestrian orientation estimation *w.r.t.* Mean Absolute Error in degree (the lower the better) and $Acc_{\frac{\pi}{8}}$,$Acc_{\frac{\pi}{4}}$ (the higher the better). | means "or". The suffix G, P, B refer to Gaussian, Poison and Binomial-uniform mixture conservative target label, respectively.

We also utilize the 36-pose labels provided in [?, ?], and adapt the backbone from [?]. We report the results *w.r.t.* mean absolute error and accuracy at $\frac{\pi}{8}$ and $\frac{\pi}{4}$ in Table 3, which are the percentage of images whose pose error is less than $\frac{\pi}{8}$ and $\frac{\pi}{4}$, respectively. Even the plain $\mathcal{L}_{d_{i,j}}(\mathrm{s},\mathrm{t})$ outperforms SHIFT [?] by 4.4% and 3.9% *w.r.t.* $Acc\frac{\pi}{8}$ and $Acc\frac{\pi}{4}$. Unfortunately, the adaptive ground metric learning is not stable when we scale the number of class to 36.

The disagreement of human labeling is significant in 36 class setting. In such a case, our conservative target label is potentially helpful. The discretized Gaussian distribution ($\xi = 0.1, \eta = 0.05, \mu = 5, \sigma^2 = 2.5$) and Binomial distribution ($\xi = 0.1, \eta = 0.05, K = 10, p = 0.5$) show similar performance, while the Poisson distribution ($\xi = 0.1, \eta = 0.05, K = 10, \lambda = 5$) appears less competitive. Note that the variance of Poisson distribution is equal to its mean λ, and it approximates a symmetric distribution with a large λ. Therefore, it is not easy to control the shape of target distribution. Our $\mathcal{L}_{\mathrm{D}^2_{i,j}}(\mathrm{s},\bar{\mathrm{t}})_B$ outperforms [?] by 6.3°, 6% and 4.9% in terms of Mean AE, $Acc\frac{\pi}{8}$ and $Acc\frac{\pi}{4}$.

Method	Mean AE	Median AE
HSSR[?]	20.30	3.36
SMMR[?]	12.61	3.52
SHIFT[?]	9.86	3.14
$\mathcal{L}_{d_{i,j}}(\mathrm{s},\mathrm{t})\vert(\mathrm{s},\bar{\mathrm{t}})$	6.46 | 6.30	2.29 | 2.18
$\mathcal{L}_{d_{i,j}}(\mathrm{s},\mathrm{t})\vert(\mathrm{s},\bar{\mathrm{t}}), t_j* = \sum_{i=0}^{N-1} s_i{}^\dagger$	6.46 | 6.30	2.29 | 2.18
$\mathcal{L}_{\mathrm{D}^2_{i,j}}(\mathrm{s},\mathrm{t})\vert(\mathrm{s},\bar{\mathrm{t}})$	6.23 | 6.04	2.15 | $\underline{2.11}$
$\mathcal{L}_{\mathrm{D}^2_{i,j}}(\mathrm{s},\mathrm{t})\vert(\mathrm{s},\bar{\mathrm{t}}), t_j* = \sum_{i=0}^{N-1} s_i{}^\dagger$	6.23 | 6.04	2.15 | $\underline{2.11}$
$\mathcal{L}_{\mathrm{D}^3_{i,j}}(\mathrm{s},\mathrm{t})\vert(\mathrm{s},\bar{\mathrm{t}})$	6.47 | 6.29	2.28 | 2.20
$\mathcal{L}_{\mathrm{D}^{H_\tau}_{i,j}}(\mathrm{s},\mathrm{t})\vert(\mathrm{s},\bar{\mathrm{t}})$	6.20 | 6.04	2.14 | 2.10

Table 4. Results on EPFL *w.r.t.* Mean and Median Absolute Error in degree (the lower the better). | means "or".† denotes we assign $t_j* = \sum_{i=0}^{N-1} s_i$, and $t_j* = 1$ in all of the other cases.

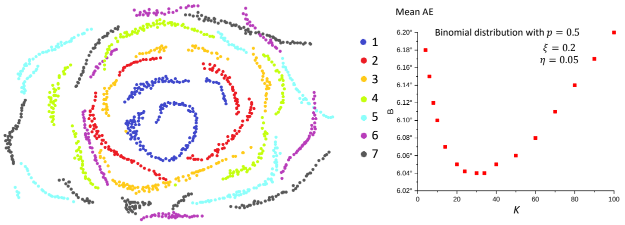


Figure 5. Left: The second-to-last layer feature of the 7 sequences in EPFL testing set with t-SNE mapping (not space position/angle). Right: Mean AE as a function of K for the Binomial distribution showing that the hyper-parameter K matters.

4.3. Vehicle orientation

The EPFL dataset [?] contains 20 image sequences of 20 car types at a show. We follow [?] to choose ResNet-101 [?] as the backbone and use 10 sequences for training and the other 10 sequences for testing. As shown in Table 4, the Huber function ($\tau = 10$) can be beneficial for noisy data learning, but the improvements appear to be not significant after we have modeled the noise in our conservative target label with Binomial distribution ($\xi = 0.2, \eta = 0.05, K = 30, p = 0.5$). Therefore, we would recommend choosing $\mathcal{L}_{\mathrm{D}^2_{i,j}}$ and Binomial-uniform mixture distribution as a simple yet efficient combination. The model is not sensitive to the possible inequality of $\sum_{i=0}^{N-1} t_i$ and $\sum_{i=0}^{N-1} s_i$ caused by numerical precision.

Besides, we visualize the second-to-last layer representation of some sequences in Fig. 5 left. As shown in Fig. 5 right, the shape of Binomial distribution is important for performance. It degrades to one-hot or uniform distribution when $K = 0$ or a large value. All of the hyper-parameters in our experiments are chosen via grid searching. We see a 27.8% Mean AE decrease from [?] to $\mathcal{L}_{\mathrm{D}^2_{i,j}}(\mathrm{s},\bar{\mathrm{t}})$, and 33% for Median AE.

4.4. 3D object pose

PASCAL3D+ dataset [?] consists of 12 common categorical rigid object images from the Pascal VOC12 and ImageNet dataset [?] with both detection and pose annotations. On average, about 3,000 object instances per category are captured in the wild, making it challenging for estimating object pose. We follow the typical experimental protocol, that using ground truth detection for both training and testing, and choosing Pascal validation set to evaluate our viewpoint estimation quality [?, ?, ?].

The pose of an object in 3D space is usually defined as a 3-tuple (azimuth, elevation, cyclo-rotation), and each of them is computed separately. We note that the range of elevation is [0,π], the Wasserstein distance for non-periodic ordered data can be computed via Eq. (9). We choose the Binomial-uniform mixture distribution ($\xi = 0.2, \eta = 0.05, K = 20, p = 0.5$) to construct our conservative label. The same data augmentation and ResNet50 from [?] (mixture of CE and regression loss) is adopted for fair comparisons, which has 12 branches for each categorical and each branch has three softmax units for 3-tuple.

We consider two metrics commonly applied in literature: Accuracy at $\frac{\pi}{6}$, and Median error (*i.e.*, the median of rotation angle error). Table 5 compares our approach with previous techniques. Our methods outperform previous approaches in both testing metrics. The improvements are more exciting than recent works. Specifically, $\mathcal{L}_{\mathrm{D}^2_{i,j}}(\mathrm{s},\bar{\mathrm{t}})$ outperforms [?] by 3.37% in terms of $Acc_{\frac{\pi}{6}}$, and the reduces $MedErr$ from 10.4 [?] to 9.37 (by 9.5%).

Besides, we further evaluate $Acc_{\frac{\pi}{18}}$, which assesses the percentage of more accurate predictions. This shows the prediction probabilities are closely distributed around the ground of truth pose.

5. Conclusions

We have introduced a simple yet efficient loss function for pose estimation, based on the Wasserstein distance. Its ground metric represents the class correlation and can be predefined using an increasing function of arc length or learned by alternative optimization. Both the outlier and inlier noise in pose data are incorporated in a unimodal-uniform mixture distribution to construct the conservative label. We systematically discussed the fast closed-form solutions in one-hot and conservative label cases. The results show that the best performance can be achieved by choosing convex function, Binomial distribution for smoothing and solving its exact solution. Although it was originally developed for pose estimation, it is essentially applicable to other

		backbone	aero	bike	boat	bottle	bus	car	chair	table	mbike	sofa	train	tv	mean
$Acc_{\frac{\pi}{6}}$	Tulsiani *et al.* [?]	AlexNet+VGG	0.81	0.77	0.59	0.93	0.98	0.89	0.80	0.62	0.88	0.82	0.80	0.80	0.8075
	Su *et al.* [?]	AlexNet†	0.74	0.83	0.52	0.91	0.91	0.88	0.86	0.73	0.78	0.90	0.86	0.92	0.82
	Mousavian *et al.* [?]	VGG	0.78	0.83	0.57	0.93	0.94	0.90	0.80	0.68	0.86	0.82	0.82	0.85	0.8103
	Pavlakos *et al.* [?]	Hourglass	0.81	0.78	0.44	0.79	0.96	0.90	0.80	N/A	N/A	0.74	0.79	0.66	N/A
	Mahendran *et al.* [?]	ResNet50†	0.87	0.81	0.64	0.96	0.97	0.95	0.92	0.67	0.85	0.97	0.82	0.88	0.8588
	Grabner *et .al.* [?]	ResNet50	0.83	0.82	0.64	0.95	0.97	0.94	0.80	0.71	0.88	0.87	0.80	0.86	0.8392
	Zhou *et al.* [?]	ResNet18	0.82	0.86	0.50	0.92	0.97	0.92	0.79	0.62	0.88	0.92	0.77	0.83	0.8225
	Prokudin *et al.* [?]	InceptionResNet	0.89	0.83	0.46	0.96	0.93	0.90	0.80	0.76	0.90	0.90	0.82	0.91	0.84
	$\mathcal{L}_{d_{i,j}}(\mathrm{s},\mathrm{t})$	ResNet50	0.88	0.79	0.67	0.93	0.96	0.96	0.86	0.73	0.86	0.91	0.89	0.87	0.8735
	$\mathcal{L}_{\mathrm{D}^2_{i,j}}(\mathrm{s},\mathrm{t})$	ResNet50	0.89	0.84	0.67	0.96	0.95	0.95	0.87	0.75	0.88	0.92	0.88	0.89	0.8832
	$\mathcal{L}_{\mathrm{D}^{H_\mathcal{T}}_{i,j}}(\mathrm{s},\mathrm{t})$	ResNet50	0.90	0.82	0.68	0.95	0.97	0.94	0.89	0.76	0.88	0.93	0.87	0.88	0.8849
	$\mathcal{L}_{\mathrm{D}^2_{i,j}}(\mathrm{s},\bar{\mathrm{t}})$	ResNet50	0.91	0.82	0.67	0.96	0.97	0.95	0.89	0.79	0.90	0.93	0.85	0.90	0.8925
$Acc_{\frac{\pi}{18}}$	Zhou *et al.* [?]	ResNet18	0.49	0.34	0.14	0.56	0.89	0.68	0.45	0.29	0.28	0.46	0.58	0.37	0.4818
	$\mathcal{L}_{d_{i,j}}(\mathrm{s},\mathrm{t})$	ResNet50	0.48	0.64	0.20	0.60	0.83	0.62	0.42	0.37	0.32	0.42	0.58	0.39	0.5020
	$\mathcal{L}_{d_{i,j}}(\mathrm{s},\bar{\mathrm{t}})$	ResNet50	0.48	0.65	0.19	0.58	0.86	0.64	0.45	0.38	0.35	0.41	0.55	0.36	0.5052
	$\mathcal{L}_{\mathrm{D}^2_{i,j}}(\mathrm{s},\mathrm{t})$	ResNet50	0.49	0.63	0.18	0.56	0.85	0.67	0.47	0.41	0.26	0.43	0.62	0.38	0.5086
	$\mathcal{L}_{\mathrm{D}^2_{i,j}}(\mathrm{s},\bar{\mathrm{t}})$	ResNet50	0.51	0.65	0.19	0.59	0.86	0.63	0.48	0.40	0.28	0.41	0.57	0.40	0.5126
	$\mathcal{L}_{\mathrm{D}^{H_\mathcal{T}}_{i,j}}(\mathrm{s},\mathrm{t})$	ResNet50	0.52	0.67	0.16	0.58	0.88	0.67	0.45	0.33	0.25	0.44	0.61	0.35	0.5108
	$\mathcal{L}_{\mathrm{D}^{H_\mathcal{T}}_{i,j}}(\mathrm{s},\bar{\mathrm{t}})$	ResNet50	0.50	0.66	0.17	0.55	0.85	0.65	0.46	0.40	0.38	0.45	0.59	0.41	0.5165
$MedErr$	Tulsiani *et al.* [?]	AlexNet+VGG	13.8	17.7	21.3	12.9	5.8	9.1	14.8	15.2	14.7	13.7	8.7	15.4	13.59
	Su *et al.* [?]	AlexNet	15.4	14.8	25.6	9.3	3.6	6.0	9.7	10.8	16.7	9.5	6.1	12.6	11.7
	Mousavian *et al.* [?]	VGG	13.6	12.5	22.8	8.3	3.1	5.8	11.9	12.5	12.3	12.8	6.3	11.9	11.15
	Pavlakos *et al.* [?]	Hourglass	11.2	15.2	37.9	13.1	4.7	6.9	12.7	N/A	N/A	21.7	9.1	38.5	N/A
	Mahendran *et al.* [?]	ResNet50	8.5	14.8	20.5	7.0	3.1	5.1	9.3	11.3	14.2	10.2	5.6	11.7	11.10
	Grabner *et al.* [?]	ResNet50	10.0	15.6	19.1	8.6	3.3	5.1	13.7	11.8	12.2	13.5	6.7	11.0	10.88
	Zhou *et al.* [?]	ResNet18	10.1	14.5	30.3	9.1	3.1	6.5	11.0	23.7	14.1	11.1	7.4	13.0	10.4
	Prokudin *et al.* [?]	InceptionResNet	9.7	15.5	45.6	5.4	2.9	4.5	13.1	12.6	11.8	9.1	4.3	12.0	12.2
	$\mathcal{L}_{d_{i,j}}(\mathrm{s},\mathrm{t})$	ResNet50	9.8	13.2	26.7	6.5	2.5	4.2	9.4	10.6	11.0	10.5	4.2	9.8	9.55
	$\mathcal{L}_{\mathrm{D}^2_{i,j}}(\mathrm{s},\mathrm{t})$	ResNet50	10.5	12.6	23.1	5.8	2.6	5.1	9.6	11.2	11.5	9.7	4.3	10.4	9.47
	$\mathcal{L}_{\mathrm{D}^{H_\mathcal{T}}_{i,j}}(\mathrm{s},\mathrm{t})$	ResNet50	11.3	11.8	19.2	6.8	3.1	5.0	10.1	9.8	11.8	9.4	4.7	11.2	9.46
	$\mathcal{L}_{\mathrm{D}^2_{i,j}}(\mathrm{s},\bar{\mathrm{t}})$	ResNet50	10.1	12.0	21.4	5.6	2.8	4.6	10.0	10.3	12.3	9.6	4.5	11.6	9.37

Table 5. Results on PASCAL 3D+ view point estimation *w.r.t.* $Acc_{\frac{\pi}{6}}$ $Acc_{\frac{\pi}{18}}$ (the higher the better) and $MedErr$ (the lower the better). Our results are based on ResNet50 backbone and without using external training data.

problems with discrete and periodic labels. In the future, we plan to develop a more stable adaptive ground metric learning scheme for more classes, and adjust the shape of conservative target distribution automatically.

6. Acknowledgement

The funding support from Youth Innovation Promotion Association, CAS (2017264), Innovative Foundation of CIOMP, CAS (Y586320150) and Hong Kong Government General Research Fund GRF (Ref. No.152202/14E) are greatly appreciated.