FisherMatch: Semi-Supervised Rotation Regression via Entropy-based Filtering

Yingda Yin Yingcheng Cai He Wang† Baoquan Chen†
Peking University

Abstract
Estimating the 3DoF rotation from a single RGB image is an important yet challenging problem. Recent works achieve good performance relying on a large amount of expensive-to-obtain labeled data. To reduce the amount of supervision, we for the first time propose a general framework, FisherMatch, for semi-supervised rotation regression, without assuming any domain-specific knowledge or paired data. Inspired by the popular semi-supervised approach, FixMatch, we propose to leverage pseudo label filtering to facilitate the information flow from labeled data to unlabeled data in a teacher-student mutual learning framework. However, incorporating the pseudo label filtering mechanism into semi-supervised rotation regression is highly non-trivial, mainly due to the lack of a reliable confidence measure for rotation prediction. In this work, we propose to leverage matrix Fisher distribution to build a probabilistic model of rotation and devise a matrix Fisher-based regressor for jointly predicting rotation along with its prediction uncertainty. We then propose to use the entropy of the predicted distribution as a confidence measure, which enables us to perform pseudo label filtering for rotation regression. For supervising such distribution-like pseudo labels, we further investigate the problem of how to enforce loss between two matrix Fisher distributions. Our extensive experiments show that our method can work well even under very low labeled data ratios on different benchmarks, achieving significant and consistent performance improvement over supervised learning and other semi-supervised learning baselines. Our project page is at https://yd-yin.github.io/FisherMatch.

1. Introduction
Incorporating deep neural networks to perform rotation regression is exerting an ever-important influence in computer vision, graphics and robotics. This is now one of the key technology in enabling a multitude of applications such as camera relocalization and visual odometry [6, 11], object pose estimation and tracking [41, 44], and 6DoF robot grasping [5, 15]. One of the major obstacles to improving rotation regression is expensive rotation annotations. Though many large-scale image datasets have been curated with sufficient semantic annotations, obtaining a large-scale real dataset with rotation annotations can be extremely laborious, expensive and error-prone [43]. With the amount of labeled data being the bottleneck, there is a demand for methods that can leverage unlabeled data.

Regarding training models with fewer labels, semi-supervised learning (SSL) has been a powerful approach, mitigating the requirement for labeled data by providing a means of leveraging unlabeled data and thus attracting more and more attention.

Recent years have witnessed many processes in semi-supervised classification [4, 12, 20, 36, 37], semi-supervised object detection [25, 40], and semi-supervised human and hand pose estimation [17, 32]. However, only few works address semi-supervised rotation regression, with most of them leveraging domain-specific knowledge, e.g., temporal smoothness of object pose [24] and strong assumptions, e.g., paired images from different viewpoints [27].

The underlying reason for little work in this field is that rotation regression is very unique and challenging. First, it is undesirable to turn the rotation regression into a classification problem. Given that 3D rotation space is continuous, discretizing the space into a small number of bins will lead to limited accuracy, which is intolerable for many applications involving rotation estimation. Also, rotation regression is even not a standard regression problem. Given that rotation space SO(3) is a non-Euclidean manifold [51], a general regression algorithm needs to be tailored, taking the nonlinear structure of the rotation space into account. This further makes semi-supervised rotation regression a more challenging and less studied topic.

In this work, for the first time, we propose a general framework, namely FisherMatch, for semi-supervised rotation regression. The problem we tackle is very general: using a neural network to regress rotation from a single RGB image. Inspired by a popular semi-supervised learning approach, FixMatch [36], initially developed for classification tasks, we attempt to process rotation regression problems in...
The key idea to the success of FixMatch is to filter out the pseudo labels with low classification confidence and only supervise the model outputs with highly confident labels. This mechanism ensures the quality of pseudo labels and thus significantly improves the performance of semi-supervised learning. The underlying assumption is that the more confident a pseudo label is, the more closed this label is to the ground truth. Or, in other words, this system needs to predict a confidence that can well indicate the correctness of its prediction. Fortunately, a classification output naturally carries the information: the probability of its prediction can be used as its prediction confidence. We argue that the availability of such a reliable confidence measure is crucial to the success of FixMatch on semi-supervised classification tasks. Similarly, when adopting FixMatch to 3D object detection, 3DIoUMatch [40] constructs a separate branch to predict the 3D IoU between the predicted bounding box and the ground truth bounding box as a localization confidence to filter out poor predictions. Although 3D IoU estimation is a regression task, 3DIoUMatch can move around the predicted bounding boxes as an augmentation trick, thus creating an infinite amount of training data for this confidence estimation module. This augmentation is crucial for such a confidence estimation module since the confidence estimation modules can only be trained using labeled data and must work on unlabeled data.

However, we argue that adopting FixMatch for rotation estimation is highly non-trivial. The biggest obstacle is how to estimate the prediction confidence for rotation regression. For rotation regression, we don’t have the largest probability from the bins as our confidence; also, for rotation estimation from a single RGB image, we can’t perform such augmentation to change our rotation prediction; yet, we still need this uncertainty estimation module to work on unlabeled data with only training on a small set of labeled data.

As pointed out by [35], probabilistic modeling of rotation is the correct way to model the uncertainty of rotation regression. Parametric statistical methods for orientation statistics have long been established [9, 16, 19, 33]. In order to better resort to SO(3) manifold which has a different topology than unconstrained values in $\mathbb{R}^N$, Deng et al. [8] and Mohlin et al. [28] incorporate Bingham distribution and matrix Fisher distribution respectively to automatically learn uncertainties along with predictions, without further supervision. Thus, such networks can provide valuable information about the quality of the prediction. We prefer matrix Fisher distribution to Bingham distribution, since its rotation representation is continuous and its loss is convex with bounded gradient magnitudes, resulting in a stable training for neural networks [22, 28].

We thus devise a matrix Fisher-based rotation regressor that takes input a single RGB image and outputs the parameter of a matrix-Fisher distribution. Given the predicted distribution, we propose to use the entropy of this distribution as a confidence measure for pseudo label filtering. Basically, only pseudo labels with high confidence, i.e., lower entropy than a threshold $\tau_{\text{entropy}}$, will pass the filtering and be used for supervising the model under training. Our experiment consistently proves that entropy is an efficient indicator of the prediction performance, not only in the case of 100 percent labeled data, but also in low data ratio cases down to 5 percent. Since FisherMatch outputs a distribution rather than a single rotation, our pseudo labels become a distribution, which requires research into the unsupervised loss enforced between two distributions. In this work, we investigate cross entropy loss and negative log likelihood loss, draw a connection between them, and find their proper usage in our experiments.

On common benchmark datasets of object rotation estimation from RGB images (ModelNet10-SO(3) and Pascal3D+) under various labeled data ratios, our experiment demonstrates a significant and consistent performance improvement over supervised learning and other semi-supervised learning baselines.

2. Related Work

Rotation regression The choice of rotation representation is one of the core issues concerning rotation regression. The commonly used representations include Euler angles, axis-angles, unit quaternions, etc. However, Euler angles suffer from gimbal lock, and quaternions have a double embedding giving rise to the existence of two disconnected local minima. Moreover, [51] argues that representations less than 4 dimensions are bound to have discontinuities and are difficult for neural networks to learn. To this end, the continuous 6D representation with Gram-Schmidt orthogonalization [51] and 9D representation with SVD orthogonalization [22] have been proposed respectively, leading to superior performance in rotation regression.

Several works propose to use probability distributions over rotations to further model prediction uncertainties along with rotation regression. In Prokudin et al. [35], parameters of a mixture of Von Mises distribution using a biternion network are estimated. Deng et al. [8] uses Bingham distribution over unit quaternions to jointly predict the rotation as well as the uncertainty. Estimation with matrix Fisher distribution [28] learns to build the probability distribution over rotation matrices with unconstrained parameters. To further express arbitrary rotation distributions and better tackle rotation regression for symmetry objects, Implicit-PDF [29] chooses to represent the distributions implicitly by neural networks, instead of distribution parameters, where the SO(3) space is uniformly discretized with the help of Hopf fibration [48].
Semi-supervised classification Semi-supervised learning is a long-studied field with a diversity of approaches, many in the field of classification. Consistency regularization and pseudo labeling are two measures with in-depth exploration. Consistency regularization was first proposed in [2] which enforces the model to predict consistently across multiple perturbations [18, 20, 37, 45, 46]. Pseudo labels [21] are artificial labels generated by the model itself and are used to further train the model, often applied along with a confidence-based thresholding to ensure the pseudo label quality. Mixmatch [4], ReMixmatch [3] and FixMatch [36] are holistic methods utilizing various augmentation and label sharpening strategies.

More recently, SimPLE [13] proposes the paired loss minimizing the statistical distance between confident and similar pseudo labels. SemCo [31] considers label semantics to prevent the degradation of pseudo label quality for visually similar classes in a co-training manner. Dash [47] and FlexMatch [49] propose dynamic and adaptive pseudo label filtering, better suited for the training process.

Semi-supervised regression Semi-supervised regression is a less-touched field compared with classification, where most of the works deal with regressing Euclidean variables, e.g., Parkinson’s disease rating scales from multiple tele-monitoring data in UCI repository [1]. Early work of CoReg [32] utilizes multiple k-nearest neighbor regressors with different distance metrics and leverages the predictions of one regressor to label the other regressors in a co-training manner. SSDKL [14] leverages the unlabeled data by minimizing predictive variance in the posterior regularization framework through the composition of neural networks and the probabilistic modeling of Gaussian processes.

Self-/semi-supervised rotation estimation Several works tackle rotation estimation in a self-supervised manner. Mustikovela et al. [30] leverages the analysis-by-synthesis technique that requires a lot of extra images for training a generative model. ViewNet [27] assumes the availability of paired data (same object, different poses). The most relevant semi-supervised learning work is NVSM [39], which shares the same assumptions on data and labels with us. In contrary to regression, NVSM builds a category-level 3D cuboid mesh with feature vectors and estimates the object rotation in a render-and-compare technique through the distance-based rotation retrieval. Less literature has been seen in the field of semi-supervised rotation regression. Mariotti et al. [26] requires paired images of an object and enforces cross-reconstruction in an analysis-and-synthesis manner via rotating the encoded neural latent variables.

Our work draws insight from both the orientation statistics and the semi-supervised learning techniques introduced above, dedicated to correlating the techniques in two well-explored fields to tackle the problem in the general setting of semi-supervised rotation regression.

3. Method

In this work, we tackle the problem of learning to predict 3D object rotation from single RGB images under a semi-supervised setting, where we have only a (small) set of labeled data \( \{ x_i, y_i \}_{i=1}^{N_l} \) and a larger set of unlabeled data \( \{ x_i^u \}_{i=1}^{N_u} \). Here, \( x^l \) and \( x^u \) represent the labeled and unlabeled RGB image respectively, and \( y^l \) represents the ground-truth rotation in \( SO(3) \) for a labeled data; \( N_l \) and \( N_u \) are the number of labeled and unlabeled images, respectively.

Following a popular semi-supervised learning approach, FixMatch [36], we adopt the teacher-student mutual learning framework, which we summarize in Section 3.1. In Section 3.2, we make use of two probabilistic models of rotation for depicting the uncertainty in rotation prediction, namely Bingham distribution and matrix Fisher distribution [8, 28], and propose to use the entropy of the predicted matrix Fisher distribution as the prediction confidence for pseudo label filtering; In Section 3.4, for the purpose of enforcing loss between the teacher and the student, we construct two loss functions between pseudo labels and predicted distributions; Finally, in Section 3.5, we introduce our training protocol in detail.

3.1. Revisit FixMatch

The teacher-student mutual learning framework is a popular approach for semi-supervised learning. Mean Teacher [37] proposes the first version, containing two jointly learned models - a teacher and a student. The parameters of the teacher model are the exponential moving average (EMA) of the student model parameters that are updated by the stochastic gradient descent. The student model is trained by the ground-truth labels for the labeled data, and for the unlabeled data, the predictions of the teacher model serve as the pseudo labels and are used to supervise the student network, through which, a history consistency is enforced between the two models.

FixMatch [36] further develops this approach by proposing two strategies: asymmetric data augmentation and confidence-based pseudo label filtering. Asymmetric data augmentation means that the teacher model is fed by weakly augmented unlabeled samples while the student model takes strongly augmented unlabeled samples which contributes to the performance gap between the teacher and the student, facilitating correct information flow to the student.

Arguably, the most important contribution of FixMatch is to demonstrate the effectiveness of confidence-based pseudo label filtering. For a non-trivial semi-supervised learning task, previous works recognize that the pseudo la-
3.2. Probabilistic Modeling of Rotation

To model the uncertainty of rotation estimation, we leverage matrix Fisher distribution to build a probabilistic model of rotation prediction, following Mohlin et al. [28].

Matrix Fisher distribution [19,34] $\mathcal{MF}(\mathbf{R}; \mathbf{A})$ is a probability distribution over $\text{SO}(3)$ for rotation matrices, whose probability density function is in the form of

$$ p(\mathbf{R}) = \mathcal{MF}(\mathbf{R}; \mathbf{A}) = \frac{1}{F(\mathbf{A})} \exp \left( \text{tr} \left( \mathbf{A}^T \mathbf{R} \right) \right) $$

(1)

where parameter $\mathbf{A} \in \mathbb{R}^{3 \times 3}$ is an arbitrary $3 \times 3$ matrix and $F(\mathbf{A})$ is the normalizing constant. The mode and dispersion of the distribution can be computed from computing singular value decomposition of the parameter $\mathbf{A}$. Assume $\mathbf{A} = \mathbf{USV}^T$ and the singular values are sorted in descending order, the mode of the distribution is computed as

$$ \hat{\mathbf{R}} = \mathbf{U} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \text{det}(\mathbf{UV}) \end{bmatrix} \mathbf{V}^T $$

(2)

and the singular values $\mathbf{S} = \text{diag}(s_1, s_2, s_3)$ indicates the strength of concentration. The larger a singular value $s_i$ is, the more concentrated the distribution is along the corresponding axis (the $i$-th column of mode $\hat{\mathbf{R}}$).

Another important probabilistic model for rotation is Bingham distribution on $S^3$ for unit quaternions. The probability density function is defined as

$$ B(q; \mathbf{M}, \mathbf{Z}) = \frac{1}{F(\mathbf{Z})} \exp \left( q^T \mathbf{M} \mathbf{Z} \mathbf{M}^T q \right) $$

(3)

where $\mathbf{M} \in O(4)$ is a $4 \times 4$ orthogonal matrix and $\mathbf{Z} = \text{diag}(0, z_1, z_2, z_3)$ is a $4 \times 4$ diagonal matrix with $0 \geq z_1 \geq z_2 \geq z_3$. The first column of parameter $\mathbf{M}$ indicates the mode and the remaining columns describe the orientation of dispersion while the corresponding $z_i$, $(i \in 1, 2, 3)$ describe the strength of the dispersion. $F(\mathbf{Z})$ is the normalizing constant.

It is well recognized that rotation matrix $\mathbf{R}$ and quaternion $\mathbf{q}$ are two different representations of rotation. Similarly, as discussed in [34], matrix Fisher distribution and Bingham distribution are equivalent to each other differing only in parameterizations and rotation representations. However, given that quaternion is not a continuous representation of rotation [51], using matrix representation to learn a deep rotation estimation model has an intrinsic advantage and usually yields better performance. [28] further shows that matrix Fisher distribution has a bounded gradient, which is favored by deep neural networks. Therefore, 9D rotation matrix is chosen as our representation, and ma-

Figure 1. Pipeline overview. Our matrix Fisher based-rotation regressor $\Phi$ takes an RGB image $x$ as input and outputs the parameter $\mathbf{A}$ of the predicted matrix Fisher distribution. We leverage a teacher-student mutual learning framework composed of a learnable student model and an exponential-moving-average (EMA) teacher model. On labeled data, the student network is trained by the ground-truth labels with the supervised loss; while on unlabeled data, the student model takes the pseudo labels from the EMA teacher. We leverage an entropy-based filtering technique to filter out noisy teacher predictions. The distribution visualization is borrowed from [28] where $x$, $y$ and $z$ shown in black axes correspond to the standard basis of $\mathbb{R}^3$, and the pdf is shown on the sphere with a jet color coding. See supplementary material Section D for details of the visualization method.

Belts generated by the teacher output suffer from significant noises [36, 40]. To this end, FixMatch proposes to filter out low-quality predictions and only supervise the student model with predictions with high confidence. This strategy avoids wrong supervision to the student model and has been proved to be very effective for challenging tasks, e.g., object detection [25, 40]. Given the difficulty of rotation regression, we further propose to leverage FixMatch as the basis of our framework for the rotation regression task.
3.3. Entropy-based Pseudo Label Filtering

Inspired by FixMatch, we only want the accurate predictions from the teacher model to “teach” the student model. Otherwise, noisy pseudo labels may slow down the training procedure, or even do harm to the whole process.

For depicting the confidence of a predicted distribution, we investigate two types of losses, cross entropy loss between two distributions, which is rarely the case for a pseudo labels are distributions, and thus we need to enforce the consistency between pseudo labels and the student outputs:

\[ L_{\text{CE}}(p_t, p_s) = H(p_t, p_s) \]  

To compute cross entropy loss between two continuous distributions, we derive the analytical formula for the cross entropy between two matrix Fisher distributions \( f \sim \mathcal{MF}(A_f) \) and \( g \sim \mathcal{MF}(A_g) \), as shown below:

Assume \( A_f = U_f S_f V_f^T \), \( A_g = U_g S_g V_g^T \), \( \gamma \) is the standard transform from unit quaternion to rotation matrix, \( e_i \) is the \( i \)-th column of \( I_4 \), and \( E_i = \gamma(e_i) \), then we can derive

\[ H(f, g) = \log F_g - \sum_{i=1}^{4} z_{gi} \left( b_i^2 + \sum_{j=1}^{4} (a_{ij}^2 - b_i^2) \frac{1}{F_j} \frac{\partial F_j}{\partial z_{fj}} \right) \]

where \( z_{gi} = \text{tr}(E_i^T S_g) \), \( z_{fj} = \text{tr}(E_i^T S_f) \),

\[ a_{ij} = \gamma^{-1}(U_f E_i V_f^T) \gamma^{-1}(U_g E_j V_g^T) \]

\[ b_i = \gamma^{-1}(U_f E_i V_f^T) \gamma^{-1}(U_f E_j V_f^T) \]

and \( F_f \) and \( F_g \) are constant wrt. parameter \( Z \). See supplementary material Section B for the derivation. Note that when \( f = g \), we can also get the entropy \( H(f) \) for matrix Fisher distribution, as shown below:

\[ H(f) = \log F_f - \sum_{i=1}^{4} \left( z_{fj} \frac{1}{F_j} \frac{\partial F_j}{\partial z_{fj}} \right) \]

3.4. Loss Function between Distributions

For the labeled set \( \{x_l^i, y_l^i\}_{i=1}^{N_l} \), we adopt the most common loss function, negative log likelihood (NLL) loss, to learn the probabilistic model of rotation, as in \([8, 28]\). This loss minimizes the negative log likelihood of the ground-truth rotation in the predicted distributions, as shown below:

\[ L_l(x^i, y^i) = -\log \left( \mathcal{MF}(y^i; A_l^i) \right) \]  

where \( A_l(x^i) \) denotes the network output fed with input \( x^i \).

For unlabeled data, both our network predictions and pseudo labels are distributions, and thus we need to enforce loss between two distributions, which is rarely the case for a regression problem. We investigate two types of losses, i.e., negative log likelihood (NLL) loss and cross entropy (CE) loss.

Cross Entropy Loss \( L_{\text{CE}} \) In classification problems, a widely-used loss function between two discrete distributions is cross entropy loss \( L_{\text{CE}} \), whose gradient is equivalent to the gradient of KL divergence between two distribution \([10]\). We thus extend cross entropy loss \( L_{\text{CE}} \) so as to enforce the consistency between pseudo labels and the student outputs:

\[ L_{\text{CE}}(p_t, p_s) = H(p_t, p_s) \]

\[ L_{\text{CE}}(p_t, p_s) = -\int_{SO(3)} \delta(y^*_t) \log p_s \text{d}R \]

where \( y^*_t \) is the mode predicted by the teacher and can be computed by SVD of \( A_t^i \) (see Section \ref{sec:3.2}).

Relationship between \( L_{\text{NLL}} \) and \( L_{\text{CE}} \) Here we intend to make connection between \( L_{\text{NLL}} \) and \( L_{\text{CE}} \). We find that \( L_{\text{CE}} \) becomes \( L_{\text{NLL}} \) when we decreases the dispersion of the distribution \( p_t \) to a Dirac distribution \( \delta(R; y^*_t) \) with its mode located at \( y^*_t \). We give a brief proof as below:

\[ L_{\text{CE}}(\text{Dirac}(p_t), p_s) = H(\delta(y^*_t), p_s) \]

\[ L_{\text{CE}}(\text{Dirac}(p_t), p_s) = -\int_{SO(3)} \delta(y^*_t) \log p_s \text{d}R \]

\[ L_{\text{CE}}(\text{Dirac}(p_t), p_s) = -\log p_s(y^*_t) = L_{\text{NLL}}(p_t, p_s). \]
This exactly resembles the label sharpening technique used in semi-supervised classification [4,36], where the teacher’s output is either sharpened or turned into a hard label. To be specific, when we turn a predicted distribution \( \mathcal{M}(A^y) \) into a hard label \( y^p \), \( L_{CE} \) becomes \( L_{NLL} \). We use \( L_{CE} \) in the experiments and investigate the different behavior of these two losses in Section 4.4.

### 3.5. Training Protocol

Our training is composed of two stages: a pre-training stage, where we train our rotation regressor on the labeled data, followed by an SSL stage where both the labeled and the unlabeled data are utilized. Our matrix Fisher-based rotation regressor is fed with an RGB image \( x \) and outputs a \( 3 \times 3 \) matrix \( A \) as the predicted parameter of the matrix Fisher distribution. We take the mode of the distribution as the predicted value.

**Pre-training** We start with a supervised training procedure on the labeled set with the supervised loss as Eq. 6. We clone the rotation regressor to obtain a pair of teacher and student networks with the same initialization, once converged.

**Semi-supervised training** In SSL stage, we utilize both the labeled data and the unlabeled data. A training batch contains a mixture of \( \{x^i\}^B_i=1 \) labeled samples and \( \{x^u_i\}^U_{i=1} \) unlabeled samples. The loss function is composed of the supervised loss applied to the labeled samples and the unsupervised loss for the unlabeled samples

\[
L = L_l(x^l, y^l) + \lambda_u L_u(x^u) \tag{11}
\]

where \( L_l \) is computed as Eq. 6, \( L_u \) is as Eq. 7, and \( \lambda_u \) is the unsupervised loss weight.

In this stage, we adopt asymmetric augmentation and an exponential-moving-average teacher as stated in Sec. 3.1.

### 4. Experiment

#### 4.1. Datasets

ModelNet10-SO(3) [23] is created by rendering 3D models of ModelNet-10 [42] that are rotated by uniformly sampled random rotations in SO(3). Following [7,28], we focus on the chair and sofa category which exhibit the least rotational symmetries in the dataset. In the experiments, we set the ratio of labeled data as 5% and 10% of the training set.

Pascal3D+ [43] contains real images from Pascal VOC and ImageNet of 12 rigid object classes. Following NVSM [39], we evaluate 6 vehicle categories (aeroplane, bicycle, boat, bus, car, motorbike) which have relatively evenly distributed poses in azimuth angles, and set the number of labeled images as 7, 20 and 50 for each category respectively. We share the same selected 7 images as NVSM such that they are spread around the pose space.

We follow the original train-test split and further divide the training split into the labeled set with ground truth and the unlabeled set without ground truth.

#### 4.2. Evaluation setup

**Baselines** To the best of our knowledge, we are the first to tackle semi-supervised rotation regression in this setting, hence the comparisons are made with self-made baselines. **Supervised-L1** uses a normal regressor and only trains on the labeled set with L1 loss with the 9D-SVD [22] rotation representation, while **Supervised-Fisher** uses our matrix Fisher regressor and also only go through the pretraining stage. As an SSL baseline, **SSL-L1-Consistency** refers to adopting FixMatch into the task with the EMA teacher and asymmetric data augmentation preserved, but only applying L1 loss as the consistency supervision between the student and teacher predictions without filtering, due to lacking the confidence measure. Here, for non-Fisher regressors, we choose L1 instead of L2 loss, as [8] points out that L1 outperforms L2 for rotation regression.

We find the most relevant work to ours is NVSM [39], which, though not regression-based, tackles the same task as ours and leverages a render-and-compare scheme through distance-based rotation retrieval. We borrow NVSM and their developed baselines as our compared baselines, including two supervised rotation estimation works (StarMap [50] and NeMo [38]) and two standard classification networks (Res50-Gene and Res50-Spec), adapted into semi-supervised learning, respectively. Due to the unavailability of the training code, we exactly follow the experiment settings of NVSM and evaluate on Pascal3D+ dataset. See supplementary Section A for more details.

**Evaluation metrics** We evaluate the experiments by the mean error, the median error (in degrees) and the accuracy within 30° between the prediction and the ground truth.

#### 4.3. Results

**Result comparison** Table 1 shows the results of our method compared with baselines on ModelNet10-SO(3) under different labeled data ratios. We can see that the results of supervised learning with the labeled data only perform similarly, regardless of using a normal or a Fisher regressor. Since these models are in fact the pre-trained models for the SSL methods in the SSL stage, their similar performance sets a common basis for a fair comparison in the SSL stage. For methods that undergo a second SSL stage, our proposed **FisherMatch** method consistently outperforms the baseline SSL method **SSL-L1-Consistency**, which demonstrates the importance of performing pseudo label filtering.

The experiment results on Pascal3D+ dataset are shown in Table 2. The results illustrate that, with the effective teacher-student mutual learning framework as well as the
Table 1. Comparing our proposed FisherMatch with the baselines on ModelNet10-SO(3) under different ratios of labeled data.

<table>
<thead>
<tr>
<th>Category</th>
<th>Method</th>
<th>5%</th>
<th>10%</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Mean↓</td>
<td>Med.↓</td>
</tr>
<tr>
<td>Sofa</td>
<td>Sup.-L1 [22]</td>
<td>44.64</td>
<td>11.42</td>
</tr>
<tr>
<td></td>
<td>Sup.-Fisher [28]</td>
<td>45.19</td>
<td>13.16</td>
</tr>
<tr>
<td></td>
<td>SSL-L1-Consist.</td>
<td>36.86</td>
<td>8.65</td>
</tr>
<tr>
<td></td>
<td>SSL-FisherMatch</td>
<td>32.02</td>
<td>7.78</td>
</tr>
<tr>
<td></td>
<td>Full Sup.</td>
<td>18.62</td>
<td>5.77</td>
</tr>
</tbody>
</table>

| Chair    | Sup.-L1 [22]            | 40.41      | 16.09      | 29.02      | 10.64      |
|          | Sup.-Fisher [28]        | 39.34      | 16.79      | 28.58      | 10.84      |
|          | SSL-L1-Consist.         | 31.20      | 11.29      | 23.59      | 8.10       |
|          | SSL-FisherMatch         | 26.69      | 9.42       | 20.06      | 7.44       |
|          | Full Sup.               | 17.38      | 6.78       | 17.38      | 6.78       |

Table 2. Comparing our proposed FisherMatch with the baselines on the 6 categories of Pascal3D+ dataset with few annotations (7, 20, 50 images). The results are averaged on 6 categories.

<table>
<thead>
<tr>
<th>Method</th>
<th>7</th>
<th>20</th>
<th>50</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Med.↓</td>
<td>Acc$_{30^\circ}$↑</td>
<td>Med.↓</td>
</tr>
<tr>
<td>Res50-Gene</td>
<td>39.1</td>
<td>36.1</td>
<td>26.3</td>
</tr>
<tr>
<td>Res50-Spec</td>
<td>46.5</td>
<td>29.6</td>
<td>29.4</td>
</tr>
<tr>
<td>StarMap [50]</td>
<td>49.6</td>
<td>30.7</td>
<td>46.4</td>
</tr>
<tr>
<td>NeMo [38]</td>
<td>60.0</td>
<td>38.4</td>
<td>33.3</td>
</tr>
<tr>
<td>NVSM [39]</td>
<td>37.5</td>
<td>53.8</td>
<td>28.7</td>
</tr>
<tr>
<td>FisherMatch</td>
<td>28.3</td>
<td>56.8</td>
<td>23.8</td>
</tr>
<tr>
<td>Full Sup.</td>
<td>8.1</td>
<td>89.6</td>
<td>8.1</td>
</tr>
</tbody>
</table>

![Graphs showing the performance metrics over training iterations.](image)

Figure 2. The visualization of the training process of SSL-FisherMatch on ModelNet10-SO(3) Sofa dataset with 5% labeled data. The four plots, from left to right and from top to bottom, show the mean errors of the predictions, the pseudo label coverage, the pseudo label quality represented by the mean errors of the pseudo labels, and the mean errors between the student model and the corresponding pseudo labels, in the process of training. All the errors are measured in degrees.

**Training process analysis** Here we show how our SSL method works during the training. In Fig. 2, the upper left plot shows that the performance of the unlabeled data increases together with the test data, which indicates the increasing quality of the teacher predictions. We can also note that the performance on the unlabeled data is slightly better than that of the test data, which is sometimes referred to as transductive semi-supervised learning.

As shown in the curves, as the SSL goes on, the improving model leads to more confident predictions indicated by the decreasing entropy and increasing pseudo label coverage, which in return fuels the learning process. The coverage of pseudo labels increases by a large margin from 40\% to the final 70\%, while the pseudo label quality still keeps stable with a shaking around 2.5\°. This indicates that entropy always acts as a good indicator of performance during the whole process. The error of the student model to the pseudo labels keeps decreasing, which further proves the effectiveness of our unsupervised loss.

4.4. Ablation Study  
**Effect of Different Unsupervised Loss and Entropy Threshold** Here, we analyze the performance of our Fish-
Figure 3. The performance of FisherMatch with CE or NLL unsupervised losses with different entropy thresholds. The experiments are done on ModelNet10-SO(3) Sofa dataset with 5% labeled data.

Figure 4. Comparison of the pseudo label coverage over training process with CE or NLL unsupervised losses and entropy thresholds. The experiments are done on ModelNet10-SO(3) Sofa dataset with 5% labeled data.

Figure 5. Visualization of the indication ability of the distribution entropy wrt. the performance. The horizontal axis is the distribution entropy and the vertical axis is the number of data points, color coded by the errors (in degrees). The experiments are done on ModelNet10-SO(3) Sofa dataset with 10% labeled data.

Table 3. Semi-supervised learning experiment based on the Bingham distribution on ModelNet10-SO(3) Sofa dataset with 10% labeled data.

<table>
<thead>
<tr>
<th>Method</th>
<th>Mean↓</th>
<th>Med↓</th>
</tr>
</thead>
<tbody>
<tr>
<td>Supervised</td>
<td>39.61</td>
<td>12.68</td>
</tr>
<tr>
<td>Sup.-Fisher</td>
<td>32.92</td>
<td>8.83</td>
</tr>
<tr>
<td>SSL-BinghamMatch</td>
<td>27.01</td>
<td>6.77</td>
</tr>
<tr>
<td>SSL-FisherMatch</td>
<td>21.29</td>
<td>5.25</td>
</tr>
</tbody>
</table>

labeled data ratio.

Comparison with Bingham-based Regressor Our designed algorithm is agnostic to the choice of the rotation representation as well as the distribution model. We further test our framework based on the Bingham distribution and report the results in Table 3.

As shown in the table, the Bingham-based framework is also able to utilize the unlabeled data and significantly improve the performance of rotation estimation. However, for both its supervised and semi-supervised version, its rotation errors are in general larger than those of matrix Fisher-based framework, since its rotation representation, quaternion, is not a continuous rotation representation, as pointed in [51], thus leading to inferior performance. See supplementary material Section A for detailed settings for SSL-BinghamMatch.

5. Conclusion and Limitations

In this paper, we tackle the problem of semi-supervised rotation regression from single RGB images in a general way. Without requiring any domain-specific knowledge or paired images, we leverage the teacher-student mutual learning framework and propose an entropy-based pseudo label filtering strategy based on the probabilistic modeling of SO(3). Our experiments demonstrate the effectiveness and advantage of our method on both ModelNet10-SO(3) and Pascal3D+ datasets.

The performance of our method may degrade when both the numbers of labeled and unlabeled data are not sufficient. In this case, the uncertainty predicted by our network can be under-estimated due to over-fitting in the small labeled data, leading to reduced effectiveness in the pseudo label filtering and thus the mutual learning.

Acknowledgements

We thank the anonymous reviewers for the insightful feedback. We would like to credit Jianran Lv from DUT for the fruitful discussions and valuable help in experiments and Yang Wang from PKU for the help in the derivation of maths. This work is supported in part by grants from the Joint NSFC-ISF Research Grant (62161146002).
References


