Graph Matching with Bi-level Noisy Correspondence

Yijie Lin\textsuperscript{1}, Mouxing Yang\textsuperscript{1}, Jun Yu\textsuperscript{2}, Peng Hu\textsuperscript{1}, Changqing Zhang\textsuperscript{3*}, Xi Peng*\textsuperscript{1}
\textsuperscript{1} Sichuan University, \textsuperscript{2} Hangzhou Dianzi University, \textsuperscript{3} Tianjin University
{linyijie.gm,yangmouxing,penghu.ml,pengx.gm}@gmail.com, yujun@hdu.edu.cn, zhangchangqing@tju.edu.cn

Abstract

In this paper, we study a novel and widely existing problem in graph matching (GM), namely, Bi-level Noisy Correspondence (BNC), which refers to node-level noisy correspondence (NNC) and edge-level noisy correspondence (ENC). In brief, on the one hand, due to the poor recognizability and viewpoint differences between images, it is inevitable to inaccurately annotate some keypoints with offset and confusion, leading to the mismatch between two associated nodes, i.e., NNC. On the other hand, the noisy node-to-node correspondence will further contaminate the edge-to-edge correspondence, thus leading to ENC. For the BNC challenge, we propose a novel method termed Contrastive Matching with Momentum Distillation. Specifically, the proposed method is with a robust quadratic contrastive loss which enjoys the following merits: i) better exploring the node-to-node and edge-to-edge correlations through a GM customized quadratic contrastive learning paradigm; ii) adaptively penalizing the noisy assignments based on the confidence estimated by the momentum teacher. Extensive experiments on three real-world datasets show the robustness of our model compared with 12 competitive baselines.

1. Introduction

Graph Matching (GM)\textsuperscript{[9, 50]} aims to establish correspondences between keypoints of different graphs, which plays an important role in various applications such as object tracking\textsuperscript{[37, 47]}, scene graph discovery\textsuperscript{[5]}, Simultaneous Localization and Mapping\textsuperscript{[4]}, and Structure-from-Motion\textsuperscript{[33]}. The key of GM is to explore and exploit the bi-level affinities between graphs, i.e., node-to-node (linear) and edge-to-edge (quadratic) affinity. For this purpose, existing methods have been devoted to integrating the bi-level information by designing advanced graph neural networks\textsuperscript{[33, 39, 40, 48]} or differentiable quadratic loss\textsuperscript{[13, 32]}. Based on the encoded high-order geometrical information, graph matching achieves promising results in correspondence estimation.

However, the success of existing GM methods highly depends on an implicit assumption, i.e., the annotated keypoint pairs are correctly associated. Unfortunately, in practice, the manual annotation process is susceptible to poor recognizability\textsuperscript{[3]} and viewpoint differences between images\textsuperscript{[26]}, which probably results in offset and even false keypoint annotations (Please refer to Fig. 4 for examples). As shown in Fig. 1, the inaccurate annotations will inevitably lead to Bi-level Noisy Correspondence (BNC)
problem. Specifically, BNC refers to node-level noisy correspondence (NNC) and accompanied edge-level noisy correspondence (ENC). As shown in Fig. 3a, BNC leads to serious performance degradation due to two potential issues. On the one hand, BNC would hinder both node-level and edge-level representation learning. Specifically, representation learning on graphs requires information propagation and aggregation between the nodes and edges, therefore, BNC will cause accumulative errors and mislead the model optimization. On the other hand, GM is subject to one-to-one mapping constraints, i.e., each keypoint in the first graph must have a unique correspondence in the second graph. Hence, “a slight move in one part may affect the situation as a whole”, i.e., one single noisy correspondence pair might result in global alignment failure and even affect the optimization of correctly annotated pairs.

To the best of our knowledge, such an essential problem has not been touched so far and it is quite challenging to solve it due to the following two reasons: i) without prior engineering like verification labels [21] for noisy correspondence, it is nearly impossible to pre-process them before training, i.e., correctly distinguish and discard the noisy correspondence in advance. Hence, it is highly expected to develop a robust matching method. ii) however, GM is not a simple one-to-many optimization problem (e.g., classification) but a many-to-many combinatorial optimization problem. One cannot solve the BNC challenge by trivially resorting to label noise learning methods [17,19,22] as they only enjoy robustness on one-to-many classification tasks.

To explore an effective solution to this challenge, we propose a novel method termed CONtrastive Matching with MOMentum distillation (COMMON). Specifically, COMMON is equipped with a robust quadratic contrastive loss that incorporates both linear and quadratic geometrical information into the contrastive learning paradigm [6,15,42]. More specifically, this study is motivated by [27] that the vanilla contrastive loss is equivalent to the linear assignment from the combinatorial optimization theory. In other words, the quadratic structure information could be incorporated into contrastive learning. Based on this motivation, we endow the contrastive loss with quadratic information through two novel graph consistency regularizers. To enhance the robustness against BNC, the proposed loss adaptively penalizes the noisy correspondence through a novel momentum distillation strategy based on the memorization effect of deep neural networks [1,45], i.e., deep networks apt to learn the simple patterns before fitting the noise. Motivated by this empirical observation, we keep a momentum version of the GM model by taking the moving average of its parameters during training. The momentum teacher model could generate high-quality pseudo-targets as additional supervision on the quadratic contrastive loss without resorting to extra verification labels. With bi-level distillation on both node and edge alignment, the proposed robust quadratic contrastive loss effectively mitigates the negative affect of BNC.

The main contributions of this work are:

- We reveal a new challenge for graph matching, termed bi-level noisy correspondence (BNC). BNC refers to the noisy correspondence on both node and edge levels, which would lead to serious performance degradation.

- To tackle the BNC challenge, we propose a robust quadratic contrastive learning loss that utilizes a novel momentum distillation strategy to rebalance the noisy assignment. The proposed contrastive loss explores the quadratic information through two simple graph consistency regularizers.

- Extensive experiments on real-world data verify the effectiveness of our method against noisy correspondence. On Pascal VOC, Spair-71k and Willow, we achieve absolute improvements of 1.6%, 1.4%, and 1.4% compared to the state-of-the-art methods.

2. Related work

In this section, we briefly introduce some recent developments in deep graph matching and contrastive learning.

2.1. Deep Graph Matching

Deep GM [11,50] aims at aligning the associated keypoints from different graphs according to node-to-node and edge-to-edge correlations. To achieve better matching effects, existing methods mainly focus on utilizing the high-order information in the graph structures. According to the choice of exploiting high-order information, most existing methods could be divided into the following two groups. i) network-designed based methods [18,24,43,48] which implicitly aggregate the high-order information through GM-customized networks. For instance, PCA [39] employs graph convolutional networks to aggregate inner-graph and cross-graph structure information, and NGM [41] proposes a matching-aware graph convolution scheme with Sinkhorn iteration [10]. ii) loss-designed based methods [13,25,32] that explicitly learn the high-order information through different differentiable quadratic loss or optimization strategies. For instance, QCDGM [13] modifies the Frank-Wolfe algorithm into a differentiable optimization scheme for quadratic constraint, and BBGM [32] proposes a differentiable combinatorial solver for quadratic assignment.

Although deep GM has achieved promising performance, almost all existing methods assume that the node-to-node and edge-to-edge correspondence is faultless and correctly associated. However, in practice, the assumption
is daunting and nearly impossible to satisfy as aforementioned in Introduction, \textit{i.e.}, BNC is inevitable due to poor annotations. Note that, although some efforts have been devoted to achieving robust GM, these methods mainly focus on the robustness against outliers \cite{29,32,33,38} and adversarial attack \cite{31}, which is remarkably different from the revealed BNC challenge. To achieve robustness against BNC, this study proposes a momentum distillation strategy that rebalances the matching loss in a data-driven way. To the best of our knowledge, this study could be the first work on BNC-robust GM.

2.2. Contrastive Learning

As one of the most effective representation learning paradigms, contrastive learning attempts to align representations of the data from different views \cite{14,27,28,44,46} by maximizing the similarity between positive pairs and minimizing that of negative pairs. The major difference between these works lies in the strategy of constructing positive and negative pairs. For example, SimCLR \cite{6} constructs positive samples through data augmentation and takes different images in the mini-batch as negatives. Moco \cite{15} proposes a memory bank module that stores a large number of negatives to increase the contrasting effect.

The differences between this study and existing works are two-fold. On the one hand, most existing methods assume that positive pairs are closely associated, which is hard to satisfy in practical applications. In contrast, our momentum distillation solution improves the robustness of contrastive learning against imperfect correspondence. On the other hand, as pointed by \cite{27}, almost all contrastive methods only consider instance discrimination problem, \textit{i.e.}, measuring the alignment between individual pairs of objects while ignoring the high-order correlation among these objects. To tackle this problem, we propose a novel quadratic contrastive learning loss that not only aligns the objects (node) but also aligns the correlation between objects (edge) inspired by \cite{27}. Different from the quadratic loss proposed in \cite{27}, our quadratic term is implemented with the mean square error instead of singular value decomposition. In addition, our quadratic term exploits the extra cross-graph quadratic information, whereas \cite{27} does not. What is more distinct, our loss is designed to achieve the cross-graph quadratic information, whereas \cite{27} does not.

3. Method

In this section, we first introduce the problem definition (Section 3.1). Then we delineate how the proposed quadratic contrastive learning loss better explores the node-to-node and edge-to-edge correlations (Section 3.2), and how the designed loss adaptively penalizes the noisy correspondence through momentum distillation (Section 3.3).

3.1. Problem Definition

For two images with \(n\) and \(m\) keypoints each (\(n \leq m\)), graph matching aims to build the node-to-node correspondence between two given graphs \(G_A\) and \(G_B\) built based on these keypoints. Let \(V_A \in \mathbb{R}^{n \times d}\) and \(V_B \in \mathbb{R}^{m \times d}\) denote feature matrix of keypoints in \(G_A\) and \(G_B\), respectively, where each row of \(V_A\) and \(V_B\) is a feature vector of a keypoint. \(F_A = V_AV_A^\top \in \mathbb{R}^{n \times n}\) and \(F_B = V_BV_B^\top \in \mathbb{R}^{m \times m}\) indicate the adjacency matrices which encode the edge information in graphs \(G_A\) and \(G_B\). Formally, graph matching seeks to solve,

\[
\min L_X (Y_{gt}, Y),
\]

where \(L_X\) measures the discrepancy between the ground-truth assignment \(Y_{gt}\) and the matching result \(Y\), \textit{e.g.}, cross-entropy loss \cite{31,41} and hamming distance loss \cite{23,32}. The matching result \(Y\) is generally obtained by,

\[
\arg \max_Y \left\{ \operatorname{tr}(Y^\top F_A Y F_B) + \operatorname{tr}(S^\top Y) \right\} \tag{2}
\]

\[\text{s.t. } Y \in \{0, 1\}^{n \times m}, \ Y1_m = 1_n, \ Y^\top 1_n \leq 1_m\]

where \(S = V_AV_B^\top \in \mathbb{R}^{n \times m}\) denotes node-to-node similarity matrix. By combining Eqs. (1) and (2), it shows that GM encourages to learn better node and edge features that leads to correct assignment.

However, due to poor recognizability and viewpoint difference of images, it is extremely hard and time-consuming to precisely annotate keypoints, leading to \textit{incorrect} assignment \(Y_{gt}\). As shown in Fig. 1 and Eq. (2), the incorrect \(Y_{gt}\) induces false matching on both node and edge levels, \textit{i.e.}, bi-level noisy correspondence. Accordingly, the BNC problem tends to cause false optimization on both node and edge levels, leading to serious performance degradation. As aforementioned, it is hard to preprocess the noisy correspondence before training. Therefore, our method attempts to explore a robust training strategy which could effectively mitigate the negative impact of BNC.

3.2. Quadratic Contrastive Learning for Graph Matching

In this section, we propose the quadratic contrastive learning loss for graph matching. Specifically, based on the combinatorial optimization theory \cite{27} that contrastive learning is equivalent to linear assignment, we accordingly introduce two graph consistency regularizers that endow contrastive learning with quadratic information.

As pointed by \cite{27}, the linear assignment problem (the second term in Eq. (2)) could be formulated by structured
linear loss [36]:

$$L = \max_{Y \in \Pi} \text{tr} \left( SY^T \right) - \text{tr} \left( SY_{gt}^T \right),$$

(3)

where \(\Pi\) denotes a set containing all \(n \times n\) permutation matrices that follows the constraint in Eq. (2). Note that \(L \geq 0\) and \(L = 0\) i.f.f. the node similarities produced by the encoder lead to the correct assignment. Through minimizing the objective in Eq. (3), the encoder expected to correctly assign keypoints from one image to another.

To make the structure linear loss calculable, one could relax the constraint \(\Pi\) with binary row-stochastic relaxation \(([Y]_{ij} \in \{0, 1\} \text{ and } \sum_j Y_{ij} = 1 \forall i)\) and utilize the common log-sum-exp approximation [2] on the max function of Eq. (3). Then the linear assignment loss could be reformulated as the InfoNCE contrastive loss as proved by [27]:

$$L = - \sum_{(i,j) \in Y_{gt}} |S|_{ij} + \tau \sum_i \log(\sum_j \exp(\frac{1}{\tau} |S|_{ij})), \quad (4)$$

where \(\tau\) controls the degree of smoothness. For our case, we reformulate the equivalence theory between the linear assignment and contrastive learning (Proposition 2 of [27]) as follows.

**Theorem 1** The log-sum-exp [2] smoothed structured linear assignment loss \(L\) with row-stochastic relaxation is equivalent to the InfoNCE contrastive loss [7, 15].

Notably, the above theorem indicates that contrastive learning Eq. (4) is a fast and differentiable approximation to the linear assignment problem thus could be efficiently used to align the keypoints in deep graph matching directly. To simply conduct contrastive learning for graph matching, we align the keypoints \(V_A\) and \(V_B\) according to \(Y_{gt}\) and only remain the keypoints which have the counterpart for training, obtaining aligned keypoints \(P_A, P_B \in \mathbb{R}^{n \times d}\) for graph \(G_A\) and \(G_B\), respectively. Then we conduct contrastive learning on both rows and columns of the node similarity matrix \(S\) as [30]:

$$L_{\text{InfoNCE}} = \mathcal{H} \left( I_n, \rho \left( P_A P_B^T \right) \right) + \mathcal{H} \left( I_n, \rho \left( P_B P_A^T \right) \right), \quad (5)$$

where \(I_n\) is the identity matrix, \(\mathcal{H}\) is the row-wise cross-entropy function with mean reduction and \(\rho\) is the softmax function applied row-wise such that each row sums to one, i.e.,

$$[\rho \left( P_A P_B^T \right)]_{ij} = \frac{\exp \left( \left[ P_A \right]_i \left[ P_B \right]_j^T / \tau \right)}{\sum_{k=1}^{n} \exp \left( \left[ P_A \right]_i \left[ P_B \right]_k^T / \tau \right)}, \quad (6)$$

where \([P_A]_i \in \mathbb{R}^{1 \times d}\) is the \(i\)-row of \(P_A\), and \(\tau\) is the softmax temperature fixed as 0.07.

Although the popular InfoNCE loss might be an effective solution to the linear assignment problem [27], it ignores another essential perspective in graph matching, i.e., edge alignment. In fact, it is generally accepted that considering the edge information in graphs makes the matching more robust [39]. Nevertheless, contrastive learning can not sufficiently utilize the graph structure and might obtain suboptimal results. To this end, we augment the contrastive loss with two novel graph-geometric consistency regularizers, namely, with-graph consistency and cross-graph consistency, to further exploit the edge correlation.
Within-graph consistency aims to encourage the alignment between edges within each image,

\[ L_{\text{within}} = \| P_A P_A^T - P_B P_B^T \|_F^2, \]  

(7)

where \( \| \cdot \|_F \) is the Frobenius norm. The within-graph consistency is a direct quadratic assignment regularizer.

Cross-graph consistency encourages the alignment of the edges across two images:

\[ L_{\text{cross}} = \| P_A P_B^T - P_B P_A^T \|_F^2. \]  

(8)

For clarity, we provide a typical example in Fig. 2 to illustrate the above formulation. Intuitively, Eq. (7) minimizes the difference between the edge in image A and its counterpart in image B, e.g., \( A_1 A_3 \) and \( B_1 B_3 \). Apart from the alignment between within-graph edges, it is also encouraged to align the cross-graph edges. Specifically, to better align the keypoints between graphs, the semantic difference between objects needs to be eliminated, e.g., the difference between two kinds of horses. Hence, given two associated keypoints and then minimize the difference between them within-graph edges by exchanging the counterpart of each.

Therefore, we establish the cross-graph edges from within-graph edges by exchanging the counterpart between each keypoint and then minimize the difference between them (e.g., \( A_1 B_3 \) and \( B_1 A_3 \)). Mathematically, Eq. (8) makes the matrix \( S \) symmetric to encourage the alignment between cross-graph edges. Although [33, 39] propose the cross-graph neural network to encode the information from both graphs, they only propagate the information from one node in image \( A \) (B) to other nodes in image \( B \) (A). In contrast, our cross-graph consistency offers a more explicit quadratic supervision to ensure the cross-graph semantic consistency.

Notably, the above two geometric consistency terms are elegantly implemented as simple regularizers on contrastive learning and facilitate the matching procedure. Accordingly, the quadratic contrastive loss for graph matching is given as:

\[ L_{\text{quadratic}} = L_{\text{InfoNCE}} + L_{\text{within}} + L_{\text{cross}}. \]  

(9)

3.3. Momentum Distillation for Robust Matching

As aforementioned, bi-level noisy correspondence would cause failure on both node-to-node and edge-to-edge matching, leading to serious performance degradation. However, different from the traditional noisy label problem, BNC is a many-to-many combinatorial optimization problem rather than a one-to-many classification problem. Therefore, it is inapplicable to distinguish [19, 21] or rectify [22] the noisy correspondence through existing noisy label classification methods.

Fortunately, Bengio et al. [1] have empirically found the memorization effect of the deep neural networks, i.e., networks apt to fit the simple patterns first. To be specific, precisely annotated correspondence could be regarded as simple patterns while noisy correspondence could be treated as complex ones. Motivated by this effect, we propose to learn from the high-quality pseudo-target generated by the momentum encoder as shown in Fig 2. Specifically, the momentum encoder is a continuously-evolving teacher, which will keep the memory at the simple patterns to some extent by taking the exponential-moving-average of the parameters from the base encoder. Formally, denoting the parameters of the base encoder as \( \theta_q \) and those of momentum encoder as \( \theta_k \), we update \( \theta_k \) by:

\[ \theta_k \leftarrow t \theta_k + (1 - t) \theta_q, \]  

(10)

where \( t \) is the momentum coefficient fixed as 0.995 in our experiments. During training, we utilize the matching prediction of the momentum encoder to rebalance the proposed quadratic contrastive learning loss. Specifically, we propose a bi-level distillation on both node and edge levels with respect to contrastive loss and graph consistency loss.

Distillation on node level. We first compute the node features \( \hat{P}_A \) and \( \hat{P}_B \) from the momentum encoder. Then we introduce a new cross-entropy term into Eq. (5) to match the alignment results between base encoder and the soft target of its momentum teacher. Formally,

\[ L_{\text{InfoNCE}} = (1 - \alpha) \left( H(I_n, \rho (\hat{P}_A P_A^T)) + H(I_n, \rho (\hat{P}_B P_B^T)) \right) + \alpha \left( H\left( \rho (\hat{P}_A P_A^T), \rho (\hat{P}_B P_B^T) \right) + H\left( \rho (\hat{P}_B P_B^T), \rho (\hat{P}_A P_A^T) \right) \right). \]  

(11)

where \( \alpha \) is the distillation temperature. The second term in Eq. (11) allows the teacher to re-calibrate the keypoint alignment by replacing the impractical perfect alignment target \( I_n \) in Eq. (5) with estimated soft-alignment probability. To keep the stability in the preliminary training stage, \( \alpha \) is linearly increasing from 0 to 0.4 in the first epoch and fixed to 0.4 afterward. As shown in Table 6, our method is insensitive to the choice of \( \alpha \).

Distillation on edge level. As the edge-level noisy correspondence is accompanied with node-level noisy correspondence, we first estimate the confidence of each node and then estimate that of the edge based on the confidence of its two vertices. Formally, let \( s \in \mathbb{R}^n \) denote the confidence of nodes where

\[ s_i = \frac{1}{2} \left( \frac{[\hat{P}_A \hat{P}_B^T]_{ii}}{\sum_j [\hat{P}_A \hat{P}_B^T]_{ij}} + \frac{[\hat{P}_B \hat{P}_A^T]_{ii}}{\sum_j [\hat{P}_B \hat{P}_A^T]_{ij}} \right). \]  

(12)

The confidence of each edge is estimated by accumulating the confidence of its two vertices, i.e., \( W = s \otimes s \) where \( \otimes \) is the outer product. Based on the edge confidence \( W \), we re-weight the graph consistency loss (Eqs. (7) and (8)).
as follows,

$$L_{\text{Rgraph}} = \|W \circ (P_A^ tp_A - P_B^ tp_B)\|_F^2 + \|W \circ (P_A^ tP_B^ t - P_B^ tP_A^ t)\|_F^2,$$  \hspace{1cm} (13)

where $\circ$ is the Hadamard product.

Different from most existing distillation methods that acquire knowledge from a pre-trained model [16, 35], our momentum distillation strategy owns the following two merits: i) enjoying the robustness against BNC by exploiting the memory effect characteristic of the momentum model, i.e., exponential-moving-average parameters would resist overfitting the noise; ii) bootstrapping the matching performance without resorting to extra knowledge or models.

Finally, by combining Eqs. (11) and (13), the overall robust quadratic contrastive learning loss is induced as,

$$L_{\text{RQuad}} = L_{\text{RInfoNCE}} + L_{\text{Rgraph}}.$$  \hspace{1cm} (14)

4. Experiments

In this section, we carry out extensive experiments on three widely-used graph matching datasets with the comparisons of 12 state-of-the-art deep graph matching approaches.

4.1. Experimental Settings

Datasets. We conduct experiments on Pascal VOC with Berkeley annotation [3], SPair-71K [26], and Willow Object Class [8]. For extensive evaluation, we report the average and per-category performance. The objective of all experiments is to maximize the average matching accuracy.

Implementation details. We implement our method in PyTorch 1.10.0 and conduct all evaluations on the Ubuntu-20.04 OS with an NVIDIA 3090 GPU. For all datasets, we use the exact same set of hyper-parameters. The encoder network consists of an ImageNet-pretrained VGG16 [34] image encoder, a graph neural network SplineCNN [12], and a two-layer projection head [6]. To optimize the networks, we adopt Adam optimizer [20] with the default parameters and set the initial learning rate as $3 \times 10^{-4}$. The learning rate for fine-tuning the VGG network is $2 \times 10^{-5}$. The batch size is set to 8 image pairs and the distillation temperature $\alpha$ is fixed to 0.4. To obtain the permutation matrix $Y$, we perform the Hungarian algorithm on the similarity matrix $S$ obtained from the base encoder following [24, 31, 39–41, 49].

Compared methods. We compare our COMMON with the following 12 popular deep graph matching methods: GMN [50], PCA [39], NGM [41], IPCA [40], CIE [48], DGMC [11], LCS [43], BBGM [32], QC-DGM [13], NGM-v2 [41], SCGM [23], and ASAR [31]. We conduct experiments for all methods on the popular open-source graph matching toolbox ThinkMatch\(^1\), for the purpose of better reproducibility and more fair comparison.

4.2. Results on Graph Matching

Pascal VOC [3] consists of 7,020 training images and 1,682 testing images with 20 classes in total. The number of nodes per graph ranges from 6 to 23. Following [32], we test our method by filtering out the non-matched points before matching. As shown in Table 1, our method outperforms the counterparts with +1.6% in terms of accuracy. It is worth pointing out that our method achieves remarkable performance improvements on the classes with prominent noisy correspondence, e.g., table (+6.8%) and sofa (+3.8%).

\(^1\)https://github.com/Thinklab-SJTU/ThinkMatch

![Image](https://example.com/figure1.png)

**Table 1.** Keypoint matching accuracy (%) on Pascal VOC with standard intersection filtering. The best and second-best results are highlighted and underlined, respectively. * denotes the results are achieved by pretraining on Spair-71k first.

<table>
<thead>
<tr>
<th>Method</th>
<th>Aero</th>
<th>Bike</th>
<th>Bird</th>
<th>Boat</th>
<th>Bottle</th>
<th>Bus</th>
<th>Car</th>
<th>Cat</th>
<th>Chair</th>
<th>Cow</th>
<th>Table</th>
<th>Dog</th>
<th>Horse</th>
<th>Mbkie</th>
<th>Person</th>
<th>Plant</th>
<th>Sheep</th>
<th>Sofa</th>
<th>Train</th>
<th>Tv</th>
<th>Mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>GMN [50]</td>
<td>41.6</td>
<td>59.6</td>
<td>60.3</td>
<td>48.0</td>
<td>79.2</td>
<td>70.2</td>
<td>67.4</td>
<td>64.9</td>
<td>39.2</td>
<td>61.3</td>
<td>66.9</td>
<td>59.8</td>
<td>61.1</td>
<td>59.8</td>
<td>37.2</td>
<td>78.2</td>
<td>68.0</td>
<td>49.9</td>
<td>84.2</td>
<td>91.4</td>
<td>62.4</td>
</tr>
<tr>
<td>PCA [39]</td>
<td>49.8</td>
<td>61.9</td>
<td>65.3</td>
<td>57.2</td>
<td>78.8</td>
<td>75.6</td>
<td>64.7</td>
<td>69.7</td>
<td>41.6</td>
<td>63.4</td>
<td>50.7</td>
<td>67.1</td>
<td>66.7</td>
<td>61.6</td>
<td>44.5</td>
<td>81.2</td>
<td>67.8</td>
<td>59.2</td>
<td>78.5</td>
<td>90.4</td>
<td>64.8</td>
</tr>
<tr>
<td>NGM [41]</td>
<td>50.1</td>
<td>63.5</td>
<td>57.9</td>
<td>53.4</td>
<td>79.8</td>
<td>77.1</td>
<td>73.6</td>
<td>68.2</td>
<td>41.1</td>
<td>66.4</td>
<td>40.8</td>
<td>60.3</td>
<td>61.9</td>
<td>63.5</td>
<td>45.6</td>
<td>77.1</td>
<td>69.3</td>
<td>65.9</td>
<td>65.5</td>
<td>92.9</td>
<td>88.2</td>
</tr>
<tr>
<td>IPCA [40]</td>
<td>53.8</td>
<td>66.2</td>
<td>67.1</td>
<td>61.2</td>
<td>80.4</td>
<td>75.3</td>
<td>72.6</td>
<td>72.5</td>
<td>44.6</td>
<td>65.2</td>
<td>54.3</td>
<td>67.2</td>
<td>67.9</td>
<td>64.2</td>
<td>47.9</td>
<td>84.4</td>
<td>70.8</td>
<td>64.0</td>
<td>83.8</td>
<td>90.8</td>
<td>67.7</td>
</tr>
<tr>
<td>LCS [43]</td>
<td>46.9</td>
<td>58.0</td>
<td>63.6</td>
<td>69.9</td>
<td>87.8</td>
<td>79.8</td>
<td>71.8</td>
<td>60.3</td>
<td>44.8</td>
<td>64.3</td>
<td>79.4</td>
<td>57.5</td>
<td>64.4</td>
<td>57.6</td>
<td>52.4</td>
<td>96.1</td>
<td>62.9</td>
<td>65.8</td>
<td>94.4</td>
<td>92.0</td>
<td>68.5</td>
</tr>
<tr>
<td>CIE [48]</td>
<td>52.5</td>
<td>68.6</td>
<td>70.2</td>
<td>57.1</td>
<td>82.1</td>
<td>77.0</td>
<td>70.7</td>
<td>73.1</td>
<td>43.8</td>
<td>69.9</td>
<td>62.4</td>
<td>70.2</td>
<td>70.3</td>
<td>66.4</td>
<td>47.6</td>
<td>85.3</td>
<td>71.7</td>
<td>64.0</td>
<td>83.9</td>
<td>91.7</td>
<td>68.9</td>
</tr>
<tr>
<td>QC-DGM [13]</td>
<td>49.6</td>
<td>64.6</td>
<td>67.1</td>
<td>62.4</td>
<td>82.1</td>
<td>79.9</td>
<td>74.8</td>
<td>73.5</td>
<td>43.0</td>
<td>68.4</td>
<td>66.5</td>
<td>67.2</td>
<td>71.4</td>
<td>70.1</td>
<td>48.6</td>
<td>92.4</td>
<td>69.2</td>
<td>70.9</td>
<td>90.9</td>
<td>92.0</td>
<td>70.3</td>
</tr>
<tr>
<td>DGMC [11]</td>
<td>50.4</td>
<td>67.6</td>
<td>70.7</td>
<td>70.5</td>
<td>87.2</td>
<td>85.2</td>
<td>82.5</td>
<td>74.3</td>
<td>46.2</td>
<td>69.4</td>
<td>69.9</td>
<td>73.9</td>
<td>73.8</td>
<td>65.4</td>
<td>51.6</td>
<td>98.0</td>
<td>73.2</td>
<td>69.6</td>
<td>94.3</td>
<td>89.6</td>
<td>73.2</td>
</tr>
<tr>
<td>BBGM [32]</td>
<td>61.9</td>
<td>71.1</td>
<td>79.7</td>
<td>79.0</td>
<td>87.4</td>
<td>94.0</td>
<td>89.5</td>
<td>80.2</td>
<td>56.8</td>
<td>79.1</td>
<td>64.6</td>
<td>78.9</td>
<td>76.2</td>
<td>75.1</td>
<td>65.2</td>
<td>98.2</td>
<td>77.3</td>
<td>77.0</td>
<td>94.9</td>
<td>93.9</td>
<td>79.0</td>
</tr>
<tr>
<td>NGM-v2 [41]</td>
<td>61.8</td>
<td>71.2</td>
<td>77.6</td>
<td>78.8</td>
<td>87.3</td>
<td>93.6</td>
<td>87.7</td>
<td>79.8</td>
<td>55.4</td>
<td>77.8</td>
<td>89.5</td>
<td>78.8</td>
<td>80.1</td>
<td>79.2</td>
<td>62.6</td>
<td>97.7</td>
<td>77.7</td>
<td>75.7</td>
<td>96.7</td>
<td>93.2</td>
<td>80.1</td>
</tr>
<tr>
<td>SCGM* [23]</td>
<td>62.9</td>
<td>72.9</td>
<td>79.6</td>
<td>79.5</td>
<td>89.3</td>
<td>94.1</td>
<td>89.1</td>
<td>79.2</td>
<td>58.4</td>
<td>79.3</td>
<td>80.5</td>
<td>79.9</td>
<td>79.5</td>
<td>76.8</td>
<td>64.8</td>
<td>98.1</td>
<td>78.0</td>
<td>75.9</td>
<td>98.0</td>
<td>93.2</td>
<td>80.5</td>
</tr>
<tr>
<td>ASAR [31]</td>
<td>62.9</td>
<td>74.3</td>
<td>79.5</td>
<td>80.1</td>
<td>89.2</td>
<td>94.0</td>
<td>88.9</td>
<td>78.9</td>
<td>58.8</td>
<td>79.8</td>
<td>88.2</td>
<td>78.9</td>
<td>79.5</td>
<td>77.9</td>
<td>64.9</td>
<td>98.2</td>
<td>77.5</td>
<td>77.1</td>
<td>98.6</td>
<td>93.7</td>
<td>81.1</td>
</tr>
</tbody>
</table>

**COMMON** | **65.6** | **75.2** | **80.8** | **79.5** | **89.3** | **92.3** | **90.1** | **81.8** | **61.6** | **80.7** | **95.0** | **82.0** | **81.6** | **79.5** | **66.6** | **98.9** | **78.9** | **80.9** | **99.3** | **93.8** | **82.7**
### Table 2. Keypoint matching accuracy (%) on SPair-71k for all classes.

<table>
<thead>
<tr>
<th>Method</th>
<th>Aero</th>
<th>Bike</th>
<th>Bird</th>
<th>Boat</th>
<th>Bottle</th>
<th>Bus</th>
<th>Car</th>
<th>Cat</th>
<th>Chair</th>
<th>Cow</th>
<th>Dog</th>
<th>Horse</th>
<th>Mbike</th>
<th>Person</th>
<th>Plant</th>
<th>Sheep</th>
<th>Train</th>
<th>Tv</th>
<th>Mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>GMN [50]</td>
<td>59.9</td>
<td>51.0</td>
<td>74.3</td>
<td>46.7</td>
<td>63.3</td>
<td>75.5</td>
<td>58.7</td>
<td>59.1</td>
<td>63.2</td>
<td>51.2</td>
<td>86.9</td>
<td>57.9</td>
<td>70.0</td>
<td>92.4</td>
<td>65.3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PCA [39]</td>
<td>64.7</td>
<td>45.7</td>
<td>78.1</td>
<td>51.3</td>
<td>63.8</td>
<td>72.7</td>
<td>61.2</td>
<td>62.8</td>
<td>62.6</td>
<td>68.2</td>
<td>59.1</td>
<td>61.2</td>
<td>64.9</td>
<td>75.7</td>
<td>87.4</td>
<td>60.4</td>
<td>72.9</td>
<td>66.0</td>
<td></td>
</tr>
<tr>
<td>NGM [41]</td>
<td>66.4</td>
<td>52.6</td>
<td>77.0</td>
<td>49.7</td>
<td>67.7</td>
<td>78.8</td>
<td>67.6</td>
<td>68.3</td>
<td>59.2</td>
<td>73.6</td>
<td>63.9</td>
<td>60.7</td>
<td>67.0</td>
<td>60.9</td>
<td>87.5</td>
<td>63.9</td>
<td>79.8</td>
<td>68.9</td>
<td></td>
</tr>
<tr>
<td>IPCA [40]</td>
<td>69.0</td>
<td>52.9</td>
<td>80.4</td>
<td>54.3</td>
<td>66.5</td>
<td>80.0</td>
<td>68.5</td>
<td>71.4</td>
<td>61.4</td>
<td>74.8</td>
<td>66.3</td>
<td>65.1</td>
<td>69.6</td>
<td>63.9</td>
<td>91.1</td>
<td>65.4</td>
<td>82.9</td>
<td>71.2</td>
<td></td>
</tr>
<tr>
<td>CIE [48]</td>
<td>71.5</td>
<td>57.1</td>
<td>81.7</td>
<td>56.7</td>
<td>67.7</td>
<td>94.7</td>
<td>87.4</td>
<td>77.4</td>
<td>72.1</td>
<td>80.7</td>
<td>74.3</td>
<td>72.5</td>
<td>79.5</td>
<td>93.4</td>
<td>98.9</td>
<td>81.2</td>
<td>94.3</td>
<td>80.2</td>
<td></td>
</tr>
<tr>
<td>NGM-v2 [41]</td>
<td>68.8</td>
<td>63.3</td>
<td>86.8</td>
<td>70.1</td>
<td>69.7</td>
<td>94.7</td>
<td>87.4</td>
<td>77.4</td>
<td>72.1</td>
<td>80.7</td>
<td>74.3</td>
<td>72.5</td>
<td>79.5</td>
<td>93.4</td>
<td>98.9</td>
<td>81.2</td>
<td>94.3</td>
<td>80.2</td>
<td></td>
</tr>
<tr>
<td>BBGM [32]</td>
<td>75.3</td>
<td>65.0</td>
<td>87.6</td>
<td>78.0</td>
<td>69.8</td>
<td>94.0</td>
<td>87.8</td>
<td>78.3</td>
<td>72.8</td>
<td>82.7</td>
<td>76.6</td>
<td>76.3</td>
<td>80.1</td>
<td>75.0</td>
<td>98.7</td>
<td>85.2</td>
<td>96.3</td>
<td>82.1</td>
<td></td>
</tr>
<tr>
<td>ASAR [31]</td>
<td>72.4</td>
<td>61.8</td>
<td>91.8</td>
<td>79.1</td>
<td>71.2</td>
<td>97.4</td>
<td>90.4</td>
<td>78.3</td>
<td>74.2</td>
<td>83.1</td>
<td>77.3</td>
<td>77.0</td>
<td>83.1</td>
<td>76.4</td>
<td>99.5</td>
<td>85.2</td>
<td>97.8</td>
<td>83.1</td>
<td></td>
</tr>
<tr>
<td>COMMON</td>
<td>77.3</td>
<td>68.2</td>
<td>92.0</td>
<td>79.5</td>
<td>70.4</td>
<td>97.5</td>
<td>91.6</td>
<td>82.5</td>
<td>72.2</td>
<td>88.0</td>
<td>80.0</td>
<td>74.1</td>
<td>83.4</td>
<td>82.8</td>
<td>99.9</td>
<td>84.4</td>
<td>98.2</td>
<td>99.8</td>
<td>84.5</td>
</tr>
</tbody>
</table>

![Figure 3](image-url)  
(a) Performance with varying noise ratios.  
(b) Initial distribution of similarity.  
(c) Similarity distribution after training.

#### 4.3. Evaluation on Noisy Correspondence

In this section, we evaluate the effectiveness of our method against noisy correspondence with synthetic noise experiments, varying viewpoint experiments, ablation studies, and parameter analysis.

#### 4.3.1 Synthetic Noise Experiment

To explicitly demonstrate the effectiveness of our method against noisy correspondence, we evaluate models on Willow Object with synthetic noisy correspondence. Concretely, we randomly select some keypoints in the training set as the noise points by adding displacement to the location coordinates. The displacement $(s, \theta)$ is generated from uniform distribution: $s \sim U(0.1, 0.2)$, $\theta \sim U(0, 360)$ where $s$ is the displacement value and $\theta$ is the angle. The displacement value is further scaled with respect to the size of the bounding box. The noisy rate $\eta$ is defined as the percentage of noise points.

#### Varying noise ratios.

As shown in Fig. 3a, we conduct experiments by varying the noise rate $\eta$ from 0 to 0.5 with an interval of 0.1. From the results, one could observe that: i) our method significantly outperforms baselines in all noise rate settings; ii) momentum distillation consistently improves the robustness against noisy correspondence.

#### Distribution of similarity scores.

We further explore the similarity scores of bi-level correspondence learned by our method with noise rate $\eta = 0.3$. In Figs. 3b and 3c, we show the histograms of similarity scores for true pairs and noisy correspondence separately. The initial similarity of these two kinds of pairs is confused and hardly distinguishable. After training, the similarity of noisy correspondence is 0.6 on average, while that of true pairs is 0.95 on average and ranges into $[0.7, 1.0]$. In other words, our method prevents noisy correspondence from dominating the network optimization, thus eliminating the negative impact of noisy correspondence.

#### 4.3.2 Evaluation with Different Viewpoint Difficulty

Spair-71k contains images with diverse variations in viewpoint and divides image pairs into easy, medium, and hard ones. In practice, we find that noisy correspondence such as occlusion is more likely to occur in hard image pairs. In other words, with the increment of viewpoint difficulty,
Figure 4. **Examples of noisy correspondence** in Pascal VOC and SPair-71k. The red line indicates wrongly annotated correspondence.

<table>
<thead>
<tr>
<th>Method</th>
<th>Car</th>
<th>Duck</th>
<th>Face</th>
<th>Mbike</th>
<th>Wbottle</th>
<th>Mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>GMN [50]</td>
<td>67.9</td>
<td>76.7</td>
<td>99.8</td>
<td>69.2</td>
<td>83.1</td>
<td>79.3</td>
</tr>
<tr>
<td>NGM [41]</td>
<td>84.2</td>
<td>77.6</td>
<td>99.4</td>
<td>76.8</td>
<td>88.3</td>
<td>85.3</td>
</tr>
<tr>
<td>PCA [39]</td>
<td>87.6</td>
<td>83.6</td>
<td><strong>100</strong></td>
<td>77.6</td>
<td>88.4</td>
<td>87.4</td>
</tr>
<tr>
<td>CIE [48]</td>
<td>85.8</td>
<td>82.1</td>
<td>99.9</td>
<td>88.4</td>
<td>88.7</td>
<td>89.0</td>
</tr>
<tr>
<td>IPCA [40]</td>
<td>90.4</td>
<td>88.6</td>
<td><strong>100</strong></td>
<td>83.0</td>
<td>88.3</td>
<td>90.1</td>
</tr>
<tr>
<td>SCGM [23]</td>
<td>91.3</td>
<td>73.0</td>
<td>100</td>
<td>95.6</td>
<td>96.6</td>
<td>91.3</td>
</tr>
<tr>
<td>ASAR [31]</td>
<td>92.5</td>
<td>84.0</td>
<td><strong>100</strong></td>
<td>95.4</td>
<td>99.0</td>
<td>94.2</td>
</tr>
<tr>
<td>LCS [45]</td>
<td>91.2</td>
<td>86.2</td>
<td><strong>100</strong></td>
<td>99.4</td>
<td>97.9</td>
<td>94.9</td>
</tr>
<tr>
<td>DGMC [11]</td>
<td><strong>98.3</strong></td>
<td>90.2</td>
<td>100</td>
<td>98.5</td>
<td>98.1</td>
<td>97.0</td>
</tr>
<tr>
<td>BBGM [32]</td>
<td>96.8</td>
<td>89.9</td>
<td><strong>100</strong></td>
<td>99.8</td>
<td><strong>99.4</strong></td>
<td>97.2</td>
</tr>
<tr>
<td>NGM-v2 [41]</td>
<td>97.4</td>
<td>93.4</td>
<td>100</td>
<td>98.6</td>
<td>98.3</td>
<td>97.5</td>
</tr>
<tr>
<td>QC-DGM [13]</td>
<td>98.0</td>
<td>92.8</td>
<td>100</td>
<td>98.8</td>
<td>99.0</td>
<td>97.7</td>
</tr>
<tr>
<td>COMMON</td>
<td><strong>97.6</strong></td>
<td><strong>98.7</strong></td>
<td><strong>100</strong></td>
<td>100</td>
<td>99.0</td>
<td><strong>99.1</strong></td>
</tr>
</tbody>
</table>

Table 3. Keypoint matching accuracy (%) on Willow Object.

<table>
<thead>
<tr>
<th>Method</th>
<th>Viewpoint difficulty</th>
<th>All</th>
</tr>
</thead>
<tbody>
<tr>
<td>BBGM [32]</td>
<td>Easy: 84.7 Medium: 78.9 Hard: 73.6 All: 82.1</td>
<td></td>
</tr>
<tr>
<td>ASAR [31]</td>
<td>Easy: 86.5 Medium: 79.1 Hard: 72.5 All: 83.1</td>
<td></td>
</tr>
<tr>
<td>COMMON</td>
<td><strong>86.6 (+0.1)</strong> Easy: <strong>81.4 (+2.3)</strong> Medium: <strong>76.4 (+2.8)</strong> Hard: <strong>84.5 (+1.4)</strong> All: <strong>82.1</strong></td>
<td></td>
</tr>
</tbody>
</table>

Table 4. Keypoint matching accuracy (%) on SPair-71k grouped by levels of difficulty in the viewpoint of the matching pair.

There might be more noisy correspondence. As shown in Table 4, our method consistently improves the matching result, particularly for the image pairs with high viewpoint difficulty (+2.8% for hard pairs). This experiment highlights the ability of our method against noisy correspondence.

### 4.3.3 Ablation Studies and Parameter Analysis

To evaluate our framework, we conduct comprehensive ablation studies by separately investigating each component. As shown in Table 5, all modules are inseparable and bring substantial performance gains. Notably, the momentum distillation strategy remarkably improves the performance by alleviating the negative impact of bi-level noisy correspondence. Besides, both two graph consistency terms play indispensable roles in learning quadratic correlations and our loss outperforms the quadratic regularization [27]. Following, we present the parameter sensitivity analysis on the distillation temperature $\alpha$ in Table 6. As shown, the distillation strategy indeed improves the performance and our method is insensitive to the choice of $\alpha$.

Table 5. **Ablation study** of COMMON in terms of accuracy. Default settings are marked in gray.

<table>
<thead>
<tr>
<th>Method</th>
<th>Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMMON (FULL)</td>
<td><strong>82.67</strong> <strong>84.54</strong></td>
</tr>
<tr>
<td>- w/o graph consistency</td>
<td>81.95 83.83</td>
</tr>
<tr>
<td>- w/o distillation</td>
<td>81.77 83.94</td>
</tr>
<tr>
<td>- InfoNCE and within-graph consistency</td>
<td>81.70 83.75</td>
</tr>
<tr>
<td>- InfoNCE and cross-graph consistency</td>
<td>81.63 83.50</td>
</tr>
<tr>
<td>- InfoNCE</td>
<td>81.33 83.38</td>
</tr>
<tr>
<td>- InfoNCE and quadratic regularization [27]</td>
<td>81.67 83.66</td>
</tr>
</tbody>
</table>

Table 6. **Parameter analysis** of COMMON with the increase of the distillation parameter $\alpha$ on Pascal VOC. Default settings are marked in gray.

<table>
<thead>
<tr>
<th>$\alpha$</th>
<th>ACC (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>81.90</td>
</tr>
<tr>
<td>0.1</td>
<td>82.34</td>
</tr>
<tr>
<td>0.2</td>
<td><strong>82.69</strong></td>
</tr>
<tr>
<td>0.3</td>
<td>82.62</td>
</tr>
<tr>
<td>0.4</td>
<td>82.67</td>
</tr>
<tr>
<td>0.5</td>
<td>82.50</td>
</tr>
<tr>
<td>0.6</td>
<td>82.59</td>
</tr>
<tr>
<td>0.7</td>
<td>82.62</td>
</tr>
<tr>
<td>0.8</td>
<td>82.66</td>
</tr>
<tr>
<td>0.9</td>
<td>82.41</td>
</tr>
</tbody>
</table>

4.4. Visualization on Noisy Correspondence

In this section, we identify some noisy correspondence pairs predicted by our model with the lowest matching similarity in Fig. 4.

5. Conclusion

This paper reveals a new problem for graph matching, i.e., bi-level noisy correspondence, which refers to wrongly annotated node-to-node correspondence and accompanied edge-to-edge correspondence. To tackle this challenge, the proposed COMMON introduces two contributions: i) a novel quadratic contrastive learning loss for graph matching, which endows contrastive learning with high-order geometrical information; ii) momentum distillation, inspired by the memorization effect of deep neural networks, which rebalances the quadratic contrastive loss to mitigate the influence of BNC. Notably, BNC might appear in not only two-graph matching problems but also multi-graph and hyper-graph matching. Our work potentially provides a novel insight into the community and may inspire some further exploration.
References


[31] Qibing Ren, Qingquan Bao, Runzhong Wang, and Junchi Yan. Appearance and structure aware robust deep visual correspondence learning via blackbox differentiation of combinatorial solvers. In NeurIPS, volume 26, 2013. 2


[45] Xiaobo Xia, Tongliang Liu, Bo Han, Chen Gong, Nannan Wang, Zongyuan Ge, and Yi Chang. Robust early-learning: Hindering the memorization of noisy labels. In ICLR, 2020. 2


[48] Tianshu Yu, Runzhong Wang, Junchi Yan, and Baoxin Li. Learning deep graph matching with channel-independent embedding and hungarian attention. In ICLR, 2019. 1, 2, 6, 7, 8
