KE-RCNN: Unifying Knowledge-Based Reasoning Into Part-Level Attribute Parsing

Xuanhan Wang, Jingkuan Song, Senior Member, IEEE, Xiaojia Chen, Lechao Cheng, Member, IEEE, Lianli Gao, Member, IEEE, and Heng Tao Shen, Fellow, IEEE

Abstract—Part-level attribute parsing is a fundamental but challenging task, which requires the region-level visual understanding to provide explainable details of body parts. Most existing approaches address this problem by adding a regional convolutional neural network (RCNN) with an attribute prediction head to a two-stage detector, in which attributes of body parts are identified from localwise part boxes. However, localwise part boxes with limit visual clues (i.e., part appearance only) lead to unsatisfying parsing results, since attributes of body parts are highly dependent on comprehensive relations among them. In this article, we propose a knowledge-embedded RCNN (KE-RCNN) to identify attributes by leveraging rich knowledge, including implicit knowledge (e.g., the attribute “above-the-hip” for a shirt requires visual/geometry relations of shirt-hip) and explicit knowledge (e.g., the part of “shorts” cannot have the attribute of “hoodie” or “lining”). Specifically, the KE-RCNN consists of two novel components, that is: 1) implicit knowledge-based encoder (IK-En) and 2) explicit knowledge-based decoder (EK-De). The former is designed to enhance part-level representation by encoding part-part relational contexts into part boxes, and the latter one is proposed to decode attributes with a guidance of prior knowledge about part-attribute relations. In this way, the KE-RCNN is plug-and-play, which can be integrated into any two-stage detectors, for example, Attribute-RCNN, Cascade-RCNN, HRNet-based RCNN, and SwinTransformer-based RCNN. Extensive experiments conducted on two challenging benchmarks, for example, Fashionpedia and Kinetics-TPS, demonstrate the effectiveness and generalizability of the KE-RCNN. In particular, it achieves higher improvements over all existing methods, reaching around 3% of AP$_{IoU+F_1}$ on Fashionpedia and around 4% of Acc$_p$ on Kinetics-TPS. Code and models are publicly available at: https://github.com/sota-joson/KE-RCNN.

Index Terms—Attribute parsing, knowledge modeling, object detection.

I. INTRODUCTION

PART-LEVEL attribute parsing refers to localize human body parts and identify their attributes within an image, in which multiple persons and countable body parts with their attributes are expected to be resolved in a unified pipeline. It is a fundamental task in computer vision, as it provides fine-grained human understanding an explainable structure of a person. Optimally addressing this task would greatly support a wide range of human-centric applications, such as human fashion analysis [1], [2], [3], [4] and human behavior analysis [5], [6], [7].

Over the past decade, we have witnessed tremendous success in instance-level recognition tasks due to the advances in regional convolution neural networks (RCNNs). Numerous RCNN-based frameworks, such as FPN [8]; Mask-RCNN [9]; Cascade-RCNN [10]; and transformer-based RCNN [11] have been developed, which have substantially pushed forward the state-of-the-art methods in object detection [8], [10], [12], [13], [14], [15], [16]; instance segmentation [9], [17]; and human parsing [18], [19], [20], [21], [22]. Inspired by this, recent attempts [4], [5], [6], [7] directly adopt RCNN-based frameworks to support part-level attribute parsing, where successful approaches are derived from object detection models by applying a new branch with attribute prediction head on part-level region features. A well-known example is the Attribute-RCNN [4] extended from Mask-RCNN [9], in which part-level attributes are identified from localwise part boxes. However, localwise part boxes with limit visual clues (i.e., part appearance only) will lead to unsatisfied parsing results, since many part-level attributes are not only decided by part-self but also relevant to others.

To handle the above issue, we argue that not only visual information derived from localwise part boxes but also relational knowledge representing rich clues of a part are needed. There are two reasons behind this: first, jointly considering part visual information with its implicit knowledge, that is, parts’ visual/geometry context, is crucial for making a correct attribute recognition. For instance, in Fig. 1, when identifying an attribute of a Sweatshirt (e.g., above-the-hip), its visual information is not sufficient enough and we also need its

Manuscript received 3 June 2022; revised 19 August 2022; accepted 22 September 2022. Date of publication 17 October 2022; date of current version 17 October 2023. This work was supported in part by the National Natural Science Foundation of China under Grant 62020106008, Grant 62122018, Grant 61772116, and Grant 61872064; and in part by the Sichuan Natural Science Foundation of China under Grant 62020106008, Grant 62020106009, and Grant 62020106010. This article was recommended by Associate Editor Z. Zeng. (Corresponding author: Jingkuan Song.)
visual contextual information (e.g., Skirt, Dress, Pants), which is a geometry relationship to another part Pants. Other attributes, such as Short or Wrist-length of a sleeve, rely on visual relationships between “sleeve” with other parts. Second, explicit knowledge is essential for understanding parts and their attributes, as it is a piece of wisdom summarized from human practice. Humans are able to identify attributes from complex situations with the help of explicit knowledge. For instance, when identifying attributes for Skirt in Fig. 1, instead of visually observing for an answer, a human tends to intuitively recall an explicit knowledge to infer a set of candidate attributes for Skirt, such as Symmetrical, Mini, and High-waist. As a result, related candidate attributes are provided, and meanwhile numerous irrelevant attributes that are associated with other parts are filtered out. In essential, imitating a human decision process is capable of providing more accurate results but not involving extra computation cost for other unrelated parts’ attributes.

Motivated by the above analysis, in this article, we aim to answer one question: how to utilize implicit/explicit knowledge to enhance part-level attribute parsing. To tackle this, we propose a knowledge-embedded RCNN (KE-RCNN), which is a simple yet effective RCNN-based framework for part-level attribute parsing. It follows an encoder-decoder design pattern that involves two novel components: 1) implicit knowledge-based encoder (IK-En) and 2) explicit knowledge embedded decoder (EK-De). Specifically, IK-En is designed to enhance part-level representation by encoding implicit knowledge about part-part relational contexts into part boxes, where it smartly decides which part-part relations are needed and what contexts to add. After that, the EK-De is proposed to identify attributes from the part-level representation with a guidance of prior knowledge about part–attribute relations, which is derived from statistical priors. With the help of the proposed knowledge modeling, KE-RCNN outperforms state-of-the-art part-level attribute parsing methods. In particular, our KE-RCNN achieves a significant improvement by around 3% of APfull of (IoU=0.5,Fβ=1) on Fashionpedia and around 4% of Accp on Kinetics-TPS.

To summarize, our main contributions are three-fold.

1) We propose an effective part-level attribute parsing method, called KE-RCNN, which identifies part-level attributes by jointly considering visual clues of a part as well as relational knowledge modeling, including implicit knowledge modeling and explicit knowledge modeling.

2) KE-RCNN is designed in a play-and-plug fashion and it can be integrated into any two-stage detectors, such as Attribute-RCNN, Cascade-RCNN, HRNet-based RCNN, and SwinTransformer-based RCNN.

3) Extensive experiments conducted on two challenging benchmarks (i.e., Fashionpedia and Kinetics-TPS) demonstrate the superiority and generalizability of our approach.

II. RELATED WORK

Part-Level Human Parsing: Traditionally, human parsing aims to segment human bodies into semantic parts. Inspired by the success of RCNN-based methods [9], [10], [12], [17], numerous frameworks for part-level human parsing have been developed, which can be categorized into bottom-up, one-stage top-down and two-stage top-down approaches. In general, bottom-up approaches [20], [23], [24], [25] interpret part-level human parsing as a parsing-then-grouping pipeline, where it first predicts instance-agnostic body parts and then groups them into corresponding human instances. Different from them, top-down approaches [18], [19], [26], [27], [28] first detect human instances and then parse each human parts independently, which becomes the mainstream solution in part-level human parsing. Furthermore, the major difference between one-stage and two-stage is whether the human detection branch is combined together with part-level RCNN in a unified manner. In a different line of part-level human parsing, recent works [4], [5], [6], [7] make one more step forward to part-level attribute parsing. In these works, they follow the traditional pipeline and adopt localwise reasoning by propagating regional visual content only, which may fail since surrounding context of a part is usually required. Instead, our method infers attribute with the help of knowledge modeling, thus improving the overall performance of part-level attribute parsing as demonstrated in Section IV.

Knowledge Modeling: Many works try to enhance deep neural networks by incorporating external knowledge. According to the formation of knowledge, these works can be classified into implicit knowledge-based methods and explicit knowledge-based methods. Note that implicit knowledge is usually stored in learned models and explicit knowledge is often summarized by human beings. In recent years, the knowledge distillation technique and self-attention mechanism are widely used in implicit knowledge-based methods [11], [29], [30], [31]. For example, some works [30], [31] build an implicit knowledge for object detection by training a high-capacity model. Then, they set this trained model as a “teacher” and enhance other object detectors with small capacity by distilling implicit knowledge from the teacher. In [11] and [32], implicit knowledge about visual context of an object...
is modeled by an attention mechanism, where only relevant contexts of the object are utilized to facilitate accurate object recognition. On the other hand, explicit knowledge-based methods adopt statistical priors as the explicit knowledge. In general, the statistical priors are often constructed from large-scale data sources (e.g., Wikipedia or Visual Genome), which record general relations among categories. Therefore, previous works cast those priors as the feature representations and encode them into deep neural networks for addressing the issue of class-imbalance, which facilitates many visual tasks, such as scene graph generation [33], [34], [35]; object detection [36]; and human parsing [19]. Though impressive, how to explore knowledge to part-level attribute parsing, still remains an open question. As a supplement to them, our method can be viewed as an early attempt to jointly explore implicit and explicit knowledge in the area of part-level attribute parsing.

III. METHODOLOGY

In this section, we first revisit standard part-level attribute parsing based on RCNNs. Then, we give a technical description of our proposed method.

A. RCNN-Based Part-Level Attribute Parsing

Given an input image \( I \), the goal of part-level attribute parsing is to localize \( N \) body parts and identify \( C \) attributes for each localized part. Traditionally, a standard pipeline utilizes a backbone network (e.g., ResNet) to project an image to a feature with a size of \( H \times W \times D \), where \( D \) indicates the number of channels and \( \{ H, W \} \) denotes the spatial size. Then, numerous region proposals are provided by applying a region proposal network [37]. Finally, a detection branch, which is a standard RCNN architecture [4] with a box classifier \( \Psi_B(\cdot) \), a location regressor \( \Psi_R(\cdot) \) as well as an attribute classifier \( \Psi_A(\cdot) \), is adopted to simultaneously locate body parts and identify attributes from the region proposals. It is worth noting that attribute parsing depends on the region proposals within the standard pipeline. However, region proposals are low-qualified bounding boxes as they are needed to further refined in RCNN. Therefore, parsing attributes conditioning on region proposals lead to numerous false predictions.

B. Knowledge-Embedded RCNN

In this section, we introduce our proposed method for part-level attribute parsing. An overview of our proposed framework is presented in Fig. 2. Different from the standard pipeline that unifies part detection and attribute parsing into one RCNN branch, our method decouples attribute parsing from the standard pipeline and establishes an independent branch, called KE-RCNN, for attribute parsing. In this setting, we first utilize a standard RCNN to refine region proposals, obtaining final detected boxes of body parts. Then, we apply the RoIAlign method [9] to extract part features from final detected boxes instead of low-qualified region proposals. To identify attributes for each detected part, our KE-RCNN first utilizes an IK-En to enhance the part feature by incorporating part-part relational contexts. Then, under a guidance of explicit knowledge about part–attribute relations, candidate attribute queries that are relevant to the part are provided. Next, conditioning on candidate attribute queries, the enhanced part feature is further projected to attribute embeddings by applying an EK-De. Finally, a calculated similarity between generated attribute embeddings and attribute queries is used to identify attributes of the part.

Notations: Before presenting details of our KE-RCNN, we give some notations for clarity. First, we denote \( v \) as a detected person and \( u \) as one of associative body parts that belong to \( v \). Their features extracted by RoIAlign are denoted as \( f_v \).
In the following, we discuss the two major components (i.e., visual and geometry context encoding) of our IK-En.

**Visual Context Encoding:** Note that each pixel-level person feature \(f_v\) represents a set of parts covering the whole person \(v\). Thus, a part feature \(f_u\) can be enhanced by incorporating its visual context relations with other parts by considering relations between \(f_u\) and \(f_v\). Following [38], we start from evenly splitting part representation \(f_u\) into two subsets, respectively, denoted as \(f_{u1} \in \mathbb{R}^{(D/2) \times S_u^1}\) and \(f_{u2} \in \mathbb{R}^{(D/2) \times S_u^2}\), which aims to enable the part representation with dynamic visual receptive fields. Furthermore, each subset has different spatial size and 1/2 number of channels compared with the person representation \(f_v\). In our encoder, \(f_{u1}\) is used to represent part visual information, while \(f_{u2}\) is further utilized to encode visual contexts by interacting with \(f_v\). Specifically, we first compute an affinity matrix by comparing \(f_{u1}\) with \(f_v\) across all spatial size. Then, each pixel feature in person representation \(f_v\) is fused into part representation \(f_{u2}\) with respect to the affinity matrix. Formally, we cast this process as follows:

\[
A = \sigma \left( (V_1^T f_v) (U f_{u2}) \right) \\
h_v = f_{u1} + (V_2^T f_v) A \\
\hat{h}_v = W_z[f_{u1}, h_v] \tag{1}
\]

where \(V_1 \in \mathbb{R}^{D \times (D/2)}\), \(V_2 \in \mathbb{R}^{D \times (D/2)}\), and \(U \in \mathbb{R}^{(D/2) \times (D/2)}\) are linear matrices that project \(f_v\) and \(f_{u1}\) into a common embedding space. \(\sigma(\cdot)\) is the standard softmax function. \(A\) is an affinity matrix, which decides what visual context in \(f_v\) is needed to propagate to the part. \(h_v \in \mathbb{R}^{(D/2) \times S_u^2}\) is an updated part feature that involves relevant visual contexts. \(W_z \in \mathbb{R}^{D \times D}\) is a learnable matrix, which linearly fuses the part’s visual information \(f_{u1}\) and visual contexts \(h_v\) to attain a visually enhanced part representation \(\hat{h}_v \in \mathbb{R}^{D \times S_u^2}\).

**Geometry Context Encoding:** In addition to visual contextual relations, recognizing attributes also benefits from geometry contexts of a part. Specifically, a geometry relation between part \(u\) and person \(v\) is encoded in their relative locations. Therefore, we represent the geometry context of the part \(u\) through

\[
h_u = W_x \left( \frac{x_u - x_v}{w_u}, \frac{y_u - y_v}{h_u}, \log \left( \frac{w_u}{w_v} \right), \log \left( \frac{h_u}{h_v} \right) \right) \tag{2}
\]

where \((x_u, y_u, w_u, h_u)\) are coordinates and scales extracted from part region, \((x_v, y_v, w_v, h_v)\) are counterpart from person region. \(W_x \in \mathbb{R}^{D \times 4}\) is a linear matrix that maps the relative geometry context into a high-dimensional vector \(h_u \in \mathbb{R}^{D \times 1}\). After that, a part representation \(f_h \in \mathbb{R}^{D \times (S_u^2 + 1)}\) with implicit knowledge (i.e., visual relation and geometry relation) is obtained by simply fusing \(h_v\) and \(\hat{h}_v\), which is formalized in

\[
f_h = h_v \oplus \hat{h}_v \tag{3}
\]

where \(\oplus\) is a concatenate operation.

### 2) Explicit Knowledge-Based Decoder:

In this section, we introduce how to identify attributes of the part by our EK-De. In particular, our key idea is to decode attributes relying on human prior knowledge, which differs from former approaches [4], [6] that directly apply an attribute classifier \(\Psi_A(\cdot)\) on part representations. Compared with the directly applying attribute classifier, decoding with human prior knowledge helps to make a correct attribute recognition, as it alleviates adverse effect from decoding irrelevant attributes.

Therefore, we embody the EK-De as a conditional projection \(\mathcal{J}(X|G, I, Q) \rightarrow \mathcal{Y}\), which maps input variable \(X\) to output variable \(\mathcal{Y}\) conditioning on \(G, I\), as well as \(Q\). For clarity, we denote \(X\) as a part representation and \(\mathcal{Y}\) as decoded attributes. \(G, I, Q\), respectively, represent explicit knowledge, part identifier and attribute queries. Next, we present details of each element.

**Explicit Knowledge \(G\):** Statistical relations between part–attribute pairs provide strong priors to infer an attribute, and it is beneficial to identifying attributes of a part. Therefore, we define explicit knowledge as an undirected relationship graph \(G : G = \langle K_P, K_A, E \rangle\), where \(K_P\) denotes part categorical nodes and \(K_A\) denotes attribute nodes, and \(E\) is a set of edges which encode all pairwise relationships between parts and attributes. Furthermore, we build this graph by calculating a frequent statistics matrix \(g \in \mathbb{R}^{N \times C}\) from the occurrence among all part–attribute pairs, where \(C\) is the number of attribute categories. Specifically, we use all relationship annotations and count frequent statistics of each part–attribute relation. After counting, each element in \(g\) is further rescaled into \((0, 1)\) by a row normalization.

**Part Identifier \(I\):** Categorical distribution \(c_u\) indicates the probability of each part classes, which is predicted from the part detection branch. For simplicity, we directly use it as the part identifier.

**Attribute Queries \(Q\):** Parameters \(W_a \in \mathbb{R}^{D \times C}\) that come from attribute classifier \(\Psi_A(\cdot)\) contain global semantic information about attribute categories since it needs to adapt to all attribute embeddings trained from all part samples. Therefore, we use \(W_a\) as initial attribute queries. Given \(g\) and \(c_u\), attribute queries are further filtered, reminding \(\hat{C}\) candidate attribute queries of the part \(u\). Formally, we cast this process as follows:

\[
c_u^a = c_u \top g \\
W_u = W_a \circ c_u^a \\
q_u = \theta(W_a^T[c_u^a]) \tag{4}
\]

where \(c_u^a \in \mathbb{R}^{1 \times C}\) denotes a weighting vector that decides which attribute is the candidate. \(\circ\) denotes the Hadamard product (elementwise broadcast multiplication) and \(W_a^T \in \mathbb{R}^{D \times C}\) is the weighted attribute queries. \(\theta(\cdot)\) denotes a filtering function that outputs \(\hat{C}\) candidate attribute queries \(q_u \in \mathbb{R}^{D \times \hat{C}}\) conditioning on \(c_u^a\), where each attribute query is selected as a candidate if its’ corresponding score in \(c_u^a\) is higher than a predefined threshold value (e.g., 0).

**Part Representation \(X\):** With all conditions, we now build part representation. Except for enhanced part feature \(f_h\) yielded from IK-En, parameters \(W_b \in \mathbb{R}^{D \times N}\) that come from box
TABLE I
COMPARISONS WITH REPRESENTATIVE METHODS ON THE FASHIONPEDIA DATASET. SYMBOL “∗” MEANS THAT BASELINE MODELS ARE REIMPLEMENTED BY US. EXPERIMENTAL RESULTS INDICATE THAT DECOUPLING ATTRIBUTE PARSING FROM PART DETECTION BRANCH SIGNIFICANTLY BENEFITS FINAL PERFORMANCES. IN MOST CASES, REPLACING STANDARD RCNN WITH THE PROPOSED KE-RCNN CAN FURTHER IMPROVE THE ATTRIBUTE PARSING PERFORMANCE BY A LARGE MARGIN.

| Without Attribute Parsing Branch | Settings | Backbone | Attribute Parsing Branch | $AP_{all}^{iou+F_1}$ | $AP_{p+att+wei}^{iou+F_1}$ | $AP_{p+att}^{iou+F_1}$ | $AP_{p}^{iou+F_1}$ | $AP_{s}^{iou+F_1}$ |
|----------------------------------|----------|----------|--------------------------|----------------------|--------------------------|----------------------|----------------|----------------|
| Attribute-RCNN [4]              | ResNet50 | -        | 26.6                     | -                    | -                        | -                    | -              | -              |
| Attribute-RCNN [4]              | ResNet101| -        | 28.6                     | -                    | -                        | -                    | -              | -              |
| Attribute-RCNN [4]              | SpineNet-49| -        | 32.4                     | -                    | -                        | -                    | -              | -              |
| Attribute-RCNN [4]              | SpineNet-96| -        | 34.0                     | -                    | -                        | -                    | -              | -              |
| Attribute-RCNN [4]              | SpineNet-143| -        | 35.7                     | -                    | -                        | -                    | -              | -              |
| Attribute-RCNN* [4]             | ResNet50 | -        | 27.3                     | 34.2                 | 8.0                      | 36.9                 | 30.4           | -              |
| Attribute-RCNN* [4]             | ResNet101| -        | 27.9                     | 35.0                 | 8.0                      | 38.1                 | 31.4           | -              |
| Cascade-RCNN* [10]             | ResNet50 | -        | 29.3                     | 37.1                 | 8.2                      | 39.0                 | 32.1           | -              |
| Cascade-RCNN* [10]             | ResNet101| -        | 29.2                     | 36.7                 | 8.6                      | 39.2                 | 31.8           | -              |
| HRNet* [18]                    | HBRNet-W18| -        | 23.7                     | 31.9                 | 8.0                      | 35.1                 | 29.1           | -              |
| HRNet* [18]                    | HBRNet-W32| -        | 27.6                     | 34.2                 | 8.6                      | 37.8                 | 31.1           | -              |
| SwinTransformer* [11]           | Swin-T   | -        | 36.2                     | 43.0                 | 18.6                     | 42.2                 | 38.0           | -              |
| SwinTransformer* [11]           | Swin-S   | -        | 47.3                     | 44.2                 | 18.9                     | 45.2                 | 39.3           | -              |

| With Attribute Parsing Branch | Settings | Backbone | Attribute Parsing Branch | $AP_{all}^{iou+F_1}$ | $AP_{p+att+wei}^{iou+F_1}$ | $AP_{p+att}^{iou+F_1}$ | $AP_{p}^{iou+F_1}$ | $AP_{s}^{iou+F_1}$ |
|--------------------------------|----------|----------|--------------------------|----------------------|--------------------------|----------------------|----------------|----------------|
| Attribute-RCNN* [4]            | ResNet50 | Standard RCNN | 35.1                     | 40.5                 | 20.5                     | 39.8                 | 36.6           | -              |
| Attribute-RCNN* [4]            | ResNet101| Standard RCNN | 35.8                     | 41.6                 | 20.5                     | 40.7                 | 37.5           | -              |
| Attribute-RCNN* [4]            | ResNet101| Standard RCNN | 39.9                     | 44.8                 | 26.6                     | 42.3                 | 40.4           | -              |
| Cascade-RCNN* [10]            | ResNet50 | Standard RCNN | 36.9                     | 42.2                 | 22.1                     | 41.9                 | 38.5           | -              |
| Cascade-RCNN* [10]            | ResNet101| Standard RCNN | 39.0                     | 45.7                 | 22.2                     | 43.1                 | 40.1           | -              |
| HRNet* [18]                   | HRNet-18 | Standard RCNN | 42.7                     | 48.8                 | 26.7                     | 44.4                 | 42.7           | -              |
| HRNet* [18]                   | HRNet-32 | Standard RCNN | 36.4                     | 40.4                 | 25.7                     | 38.1                 | 36.3           | -              |
| SwinTransformer* [11]          | Swin-T   | Standard RCNN | 35.2                     | 39.5                 | 22.8                     | 39.9                 | 36.3           | -              |
| SwinTransformer* [11]          | Swin-S   | Standard RCNN | 39.0                     | 43.1                 | 27.0                     | 41.7                 | 39.7           | -              |

where $f \in \mathbb{R}^{D \times C}$ is the decoded attribute embeddings. MSA(query, key, value) denotes the standard multihead self-attention function that decodes value depending on query and key. LN is the LayerNorm function for normalizing input feature and MLP is the multilayer perceptron that applies non-linear transformation on input features. It is worth noting that in the original transformer, both query and key are derived from the same inputs. However, in our model, they are derived from two different representations (i.e., $q_u$ and $f$), where the former one is dynamically produced according to the particular part $u$. Compared with the original version, our model is more flexible and reliable to produce accurate attributes since numerous irrelevant attribute embeddings are removed by (4).

Next, we define $\psi(\cdot)$ as a similarity measurement based on the Euclidean distance. Therefore, attributes of the part are identified through a similarity matrix calculated between $q_u$ and $f$, as formalized in

$$O = \mathcal{P} \left( \sum_{i=1}^{D} q_{i} \odot f^{i} \right)$$

where $\mathcal{P}(\cdot)$ is the Sigmoid nonlinear function. $O \in \mathbb{R}^{C}$ is the attribute categorical distribution, where each element indicates the predicted probability of attribute category.
IV. EXPERIMENT

In this section, we perform extensive experiments on two challenging benchmarks: one part-level fashion parsing dataset (i.e., Fashionpedia) and one part-level action parsing dataset (i.e., Kinetics-TPS). In the following, we first introduce implementation details. Then, we compare the proposed KE-RCNN with the previous state of the arts on the two tasks. Next, we perform extensive ablation studies to explore the importance design of KE-RCNN.

A. Implementation Details

The KE-RCNN is implemented based on OpenMMLab\(^1\) on an Ubuntu server with eight Tesla V100 graphic cards. We adopt the FPN, which is pretrained on ImageNet, as the backbone model unless otherwise stated. We separately train models on the two aforementioned datasets with their respective annotations. Following common practice used in previous works [8], [9], [11], we use the SGD solver for optimizing the convolution-based model and Adam solver for the Transformer-based model. When training model on the Fashionpedia dataset, the learning rate is 1e-4 and it is decreased by 10 at the 28th and 30th epochs. Besides, the training is stopped at 32th epoch. For Kinetics-TPS, we train for 12 epochs, starting from a learning rate of 0.02 and decreasing it by 10 at the 8th and 11th epochs. A batch size of 16 is used. To provide full details of our approach, our code is made publicly available.

B. Part-Level Fashion Parsing on Fashionpedia

Dataset and Metrics: The Fashionpedia dataset [4] is used for evaluating part-level fashion parsing models. It contains 48k images in total, which are collected from Flickr and free license photo websites. It is divided into two subsets: 1) 45,623 images for training and 2) 11,585 images for validation, respectively. The part-level annotations cover 46 apparel categories, for example, dress, shorts, leg warmer, and involve 294 attributes, for example, fit, above-the-knee, regular. Following official settings [4], we adopt AP\(_{\text{IoU+I-F1}}\) to evaluate the performance. It is an extended version of standard detection metric defined in COCO [40], which considers both IoU score for the detected part and macro F1 score for the predicted attributes of the detected part. Based on this, we report standard mean average precision over the validation set: 1) AP\(_{\text{IoU+I-F1}}\) (the mean of box AP scores across all IoU thresholds (ranging from 0.5 to 0.95), all macro F1 scores, and all apparel categories); 2) AP\(_{\text{IoU+I-F1}}\) for outerwear categories; 3) AP\(_{\text{IoU+I-F1}}\) for garment parts categories; 4) AP\(_{\text{IoU+I-F1}}\) (the mean of box AP scores across all IoU thresholds and all apparel categories with a F1 threshold equal to 0.5); and 5) AP\(_{\text{IoU+I-F1}}\) (the mean of box AP scores across all IoU thresholds and all apparel categories with a F1 threshold equal to 0.75).

Main Results: We compare our approach with the state-of-the-art attribute parsing approaches on the Fashionpedia validation set. Specifically, we choose four representative RCNN-based models as baselines, including Attribute-RCNN [4], Cascade-RCNN [10], HRNet-based RCNN [18], and SwinTransformer-based RCNN [11]. In particular, we implement two versions for each baseline model. In the first version, we unify part detection and attribute parsing into one RCNN branch, which is the same as traditional methods [4]. In the second version, we build an independent RCNN branch for attribute parsing, where it identifies attributes from refined detected boxes rather than region proposals. Following common practices [4], [9], we adopt a standard RCNN as the attribute parsing branch in baseline model, where it consists of four consecutive convolution layers followed by two fully connected layers. Table I lists several standard evaluation metrics for different method/backbone pairs. For fair comparisons with baseline models, we report our reimplementation results of them, which are comparable to or higher than those were reported in papers.

From the results, we find that all traditional approaches based on the standard parsing pipeline are significantly improved after decoupling attribute parsing from standard pipeline, where overall improvements achieve at least 6% AP\(_{\text{IoU+I-F1}}\). For example, the Attribute-RCNN with proposed KE-RCNN outperforms first version of baselines by 11.8%–12.0% AP\(_{\text{IoU+I-F1}}\) when applying different backbones (i.e., ResNet-50 and ResNet-101). On two higher baselines of 29.3% AP\(_{\text{IoU+I-F1}}\) using Cascade-RCNN and 36.2% AP\(_{\text{IoU+I-F1}}\) using SwinTransformer-RCNN.
TABLE IV
ABLATION STUDY ON FASHIONPEDIA. INVESTIGATING THE EFFECT OF IK-EN VARIANTS

| Part representation | $A_{P_{all}}^{IoU+F_1}$ | $A_{outerwear}^{IoU+F_1}$ | $A_{parts}^{IoU+F_1}$ |
|---------------------|--------------------------|-----------------------------|----------------------|
| $\{h_x, h_x, f_c^2\}$ | 39.1                     | 44.2                        | 26.0                 |
| $\{h_x, h_x\}$      | 38.5                     | 43.0                        | 26.3                 |
| $\{h_x\}$           | 38.2                     | 42.4                        | 26.5                 |

TABLE V
ABLATION STUDY ON THE FASHIONPEDIA DATASET. INVESTIGATING THE EFFECT OF PART REPRESENTATION

| Part representation | $A_{P_{all}}^{IoU+F_1}$ | $A_{outerwear}^{IoU+F_1}$ | $A_{parts}^{IoU+F_1}$ |
|---------------------|--------------------------|-----------------------------|----------------------|
| $\{h_x, h_x, f_c^2\}$ | 39.1                     | 44.2                        | 26.0                 |
| $\{h_x, h_x\}$      | 38.5                     | 43.0                        | 26.3                 |
| $\{h_x\}$           | 38.2                     | 42.4                        | 26.5                 |

TABLE VI
INVESTIGATING THE EFFECT OF PRIOR KNOWLEDGE

| Modifications       | $A_{P_{all}}^{IoU+F_1}$ | $A_{outerwear}^{IoU+F_1}$ | $A_{parts}^{IoU+F_1}$ |
|---------------------|--------------------------|-----------------------------|----------------------|
| Averaging           | 37.9                     | 42.5                        | 25.1                 |
| Statistics (Fashionpedia) | 39.1                     | 44.2                        | 26.0                 |
| Statistics (Wikipedia) | 39.6                     | 44.3                        | 27.0                 |

$A_{P_{all}}^{IoU+F_1}$ using the Swin-T framework, the gains by KE-RCNN are also high, achieving +11.9% $A_{P_{all}}^{IoU+F_1}$ and +5.9% $A_{P_{all}}^{IoU+F_1}$, respectively. It indicates that the performance of attribute parsing increases as identifying attributes from refined boxes, rather than from low-qualified region proposals. The comparison results between two versions of baseline models also support this fact. This is reasonable since attribute parsing depends on part detection. When comparing the KE-RCNN with baseline models based on the second version, experimental results in Table I also show consistent improvements (e.g., around 3% $A_{P_{all}}^{IoU+F_1}$) by KE-RCNN at various evaluation metrics, indicating the effectiveness and generalizability of the proposed method. Furthermore, it also demonstrates that implicit and explicit knowledge modeling is a promising direction for attribute parsing.

C. Part-Level Action Parsing on Kinetics-TPS

Dataset and Metrics: For part-level action parsing, we use the Kinetics-TPS dataset\(^2\) for model evaluation. It contains 3809 videos in total, which are collected from a subset of the Kinetics dataset [41]. We randomly pick 30% of training set as the validation set, resulting in 2686 videos for training and 1123 videos for validation. Following official setting, we report several metrics on the validation set: 1) ACC\(_p\) (the mean of video classification accuracy conditioned on frame-level part state correctness (PSC)); 2) ACC\(_s\) (the mean of part-level action classification accuracy); and 3) AP\(_{box}\) (the mean of box AP scores across all body part categories).

Main Results: Similar to experiments conducted on Fashionpedia, we adopt Attribute-RCNN, Cascade-RCNN, HRNet-based RCNN, and SwinTransformer-based RCNN as the baseline models. Corresponding results are shown in Table II. In line with the findings from Table I, the proposed KE-RCNN outperforms baselines by a margin (+2.2~4.4 ACC\(_p\)). Based on this, one can conclude that our method performs general improvement on the part-level action parsing problem.

D. Ablation Study

To deeply analyze the proposed method and its components, we conduct extensive ablation studies on the Fashionpedia dataset. We choose the decoupling version of Attribute-RCNN with ResNet50 backbone as the baseline model. In the following, we first conduct an ablation study to investigate effects of each proposed component. Then, we would like to attain a further insight into implicit knowledge modeling as well as explicit knowledge. After that, we provide a deep analysis of learned models from a visualization aspect.

Ablation Studies of Each Component: The KE-RCNN consists of IK-En and EK-De. In this section, we would like to investigate the effect of each component for attribute parsing. Based on this, we build two additional variants of KE-RCNN, where each consists of either IK-En or EK-De.

---

\(^2\)https://deeperaction.github.io/kineticstps/
Fig. 5. Investigating what visual contexts KE-RCNN utilizes to identify attributes. The first row (a) presents visual contexts for parsing fashion attributes, while the second row (b) displays visual contexts for parsing action attributes. For each example, it visualizes the affinity matrix $A$, since it decides which part of a person feature is used to identify attributes.

Fig. 6. Qualitative comparison. From the left to right: The input images, ground truth part-level attributes, predictions from Attribute-RCNN [4], Cascade-RCNN [10], Swin-T [11], and ours, respectively. Color green denotes corrected attribute parsing by model and color red denotes failure cases. Each part is drawn in a distinguished color. Zoom in for a better view.

The experimental results are reported in Table III, where all models are tested on two benchmarks (i.e., Fashionpedia and Kinetics-TPS). From the results, we have following findings.

1) Both KE-RCNN with IK-En and KE-RCNN with EK-De outperform standard RCNN on both Fashionpedia validation set (35.1% versus 37.2% and 35.1% versus 38.4%) and the Kinetics-TPS validation set (49.1% versus 52.2% and 49.1% versus 51.7%), suggesting that incorporating implicit or explicit knowledge facilitates attribute parsing.

2) The KE-RCNN with EK-De shows a better performance (i.e., 37.2% versus 38.4%) than that of KE-RCNN with IK-En for fashion attribute parsing. However, the results are reversed (i.e., 52.2% versus 51.7%) when applying them for action attribute parsing. This suggests that identifying dynamic attributes (e.g., action state) benefits more from implicit knowledge than that from explicit knowledge, while identifying static attributes (e.g., fashion tags) prefers explicit knowledge.

3) Jointly applying IK-En and EK-De brings the best results on both two tasks, suggesting that each component is complementary to each other for attribute parsing.

Ablation Studies of Part Extraction: The global context of a part is the key to attribute parsing. Hence, we investigate various options of part context modeling and compare three different RCNN variants, including:

1) “Standard RCNN,” where the attribute parsing branch consists of four consecutive
convolution layers for generating the part representation; 2) “ASPP RCNN,” which refers to inserting multiple dilated convolutions into “Standard RCNN” for enlarging receptive fields of bounding boxes; 3) IK-En, which denotes the KE-RCNN is composed of proposed IK-En only. In particular, we choose ASPP RCNN as one of the options due to its effectiveness of contextual modeling on bounding boxes, as demonstrated in [42]. For fair comparison, all models adopt one fully connected layer to predict attributes for a part.

From experimental results summarized in Table IV, we observe that replacing standard RCNN with ASPP RCNN brings minor gains but requires a large model size. It suggests that contextual modeling by enlarging visual receptive field has a little effect to attribute parsing, and simply enlarging model capacity has reached performance bottleneck as well. Second, the proposed IK-En outperforms standard RCNN by 2.1 APall. It further drops to 38.2% after model capacity has reached performance bottleneck as well. From experimental results summarized in Table IV, we observe that replacing standard RCNN with ASPP RCNN brings minor gains but requires a large model size. It suggests that contextual modeling by enlarging visual receptive field has a little effect to attribute parsing, and simply enlarging model capacity has reached performance bottleneck as well. Second, the proposed IK-En outperforms standard RCNN by 2.1 APall. It further drops to 38.2% after model capacity has reached performance bottleneck as well. From the results, we can find that the explicit knowledge is critical, as the performance decreases by 1.2% (37.9% versus 39.1%) if the explicit knowledge is removed (see 1st and 2nd settings). Furthermore, using explicit knowledge stored in Wikipedia improves the performance from 39.1 to 39.6. This suggests that explicit knowledge is particularly effective for accurate attribute parsing.

Analysis of Learned Model: As demonstrated in [4], current attribute parsing models trained on imbalanced datasets are likely dominated by major classes. As shown in Fig. 3, the number of labeled samples for outerwear in Fashionpedia are much larger than that of garment part, resulting in a large performance gap (e.g., 26% AP) between them as shown in Fig. 4. To investigate whether the proposed KE-RCNN suffers from this or not, we evaluate them on “hard” cases. In particular, these hard cases are selected from samples of tail classes. We report evaluation results in Table VII. From the results, we observe that our method significantly improves two baselines (+7.3 and +5.0) for parsing those hard classes, indicating that the adverse effect caused by imbalanced issue can be alleviated with the proposed method. It is worth noting that the proposed KE-RCNN identifies attributes of a part with the help of general explicit knowledge, thus improving performance for hard examples.

To further attain insight into the learned model, we visualize the implicit knowledge that the KE-RCNN utilizes to parse attributes of a part. In particular, the affinity matrix \( \mathcal{A} \) for each part is visualized, since it decides which part of a person feature is used to identify attributes. The visualization results in Fig. 5 show that the KE-RCNN can well focus on relevant contexts for parsing attributes of a part. For example, when identifying “above-the-hip” for the Jacket, the KE-RCNN focuses on both torso and hip. In terms of action attributes, relevant body parts that perform actions are also captured by the KE-RCNN as well.

Qualitative Comparison: In Fig. 6, we observe that the KE-RCNN identifies attributes of a part with the help of general explicit knowledge, and thus improving performance for hard examples. It is worth noting that the proposed KE-RCNN identifies attributes of a part with the help of general explicit knowledge, and thus improving performance for hard examples.
method performs the best while the Attribute-RCNN performs the worst. In addition, given six parts examples, our method provides four parts with completely correct attribute predictions, while the most competitive method Swin-T provides three parts with completely right attributes. To summary, this qualitative results are consistent with the quantitative results shown in Table I, which demonstrates the superiority of our proposed method and proves the important role of implicit and explicit knowledge in part-level attribute parsing.

The Bottleneck of KE-RCNN: The part-level attribute parsing depends on the precise predictions of subtasks, that is, the body detection, the body part detection and the prediction of attributes. In this section, we perform an experiment to gauge the relative difficulty of subtasks, that is, which part is the main bottleneck of part-level attribute parsing so far. We evaluate KE-RCNN on the Fashionpedia dataset as well as the Kinetics-TPS dataset, hoping to inspire future research. Specifically, we replace predictions with corresponding ground-truth labels. The results in Table VIII suggest that there is still a large room for improvement in part detection and attribute parsing.

In addition, Fig. 7 shows visualization results obtained from predictions for hard examples. From Fig. 7, one can conclude that hard examples share a common characteristic of “small.” Intuitively, parsing those hard examples requires rich visual contexts. The proposed KE-RCNN jointly encodes such contexts by implicit knowledge modeling as well as explicit knowledge modeling, thus improving overall performance for those hard examples. For more details, we refer the reader to supplementary materials.

Fig. 7. Qualitative comparison for parsing “hard” cases, involving Collar, Lapel, Sleeve, Pocket, and Neckline. Color green denotes corrected attribute parsed by model and color red denotes failure cases.

The Bottleneck of KE-RCNN: The part-level attribute parsing depends on the precise predictions of subtasks, that is, the body detection, the body part detection and the prediction of attributes. In this section, we perform an experiment to gauge the relative difficulty of subtasks, that is, which part is the main bottleneck of part-level attribute parsing so far. We evaluate KE-RCNN on the Fashionpedia dataset as well as the Kinetics-TPS dataset, hoping to inspire future research. Specifically, we replace predictions with corresponding ground-truth labels. The results in Table VIII suggest that there is still a large room for improvement in part detection and attribute parsing.

In addition, Fig. 7 shows visualization results obtained from predictions for hard examples. From Fig. 7, one can conclude that hard examples share a common characteristic of “small.” Intuitively, parsing those hard examples requires rich visual contexts. The proposed KE-RCNN jointly encodes such contexts by implicit knowledge modeling as well as explicit knowledge modeling, thus improving overall performance for those hard examples. For more details, we refer the reader to supplementary materials.

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
V. CONCLUSION AND DISCUSSION

In this article, we proposed an effective method, called KE-RCNN, aiming at identifying part-level attributes by utilizing implicit and explicit knowledge. By building IK-En, we enhanced part representations by incorporating visual contexts as well as geometry contexts. Then, EK-De was proposed to identify attributes of a part by human prior knowledge. Extensive experiments on two benchmarks proved the effectiveness and generalizability of our approach.

REFERENCES

[1] N. Inoue, E. Simo-Serra, T. Yamasaki, and H. Ishikawa, “Multi-label fashion image classification with minimal human supervision,” in Proc. ICCVW, 2017, pp. 2261–2267.
[2] U. Mall, K. Matzen, B. Hariharan, N. Snavely, and K. Bala, “GeoStyle: Discovering fashion trends and events,” in Proc. ICCV, 2019, pp. 411–420.
[3] Y. Ge, R. Zhang, X. Wang, X. Tang, and P. Luo, “DeepFashion2: A versatile benchmark for detection, pose estimation, segmentation and re-identification of clothing images,” in Proc. CVPR, 2019, pp. 5337–5345.
[4] M. Jia et al., “Fashionpedia: Ontology, segmentation, and an attribute localization dataset,” in Proc. ECCV, 2020, pp. 316–332.
[5] X. Chen, X. Liu, K. Liu, W. Liu, and T. Mei, “A baseline framework for part-level action parsing and action recognition,” 2021, arXiv:2110.03368.
[6] Y.-L. Li et al., “PaStaNet: Toward human activity knowledge engine,” in Proc. CVPR, 2020, pp. 379–388.
[7] X. Wang, X. Chen, L. Gao, B. Lin, and J. Song, “Technical report: Disentangled action parsing networks for accurate part-level action parsing,” 2021, arXiv:2111.03225.
[8] T. Lin, P. Dollár, R. B. Girshick, K. He, B. Hariharan, and S. J. Belongie, “Feature pyramid networks for object detection,” in Proc. CVPR, 2017, pp. 935–944.
[9] K. He, G. Gkioxari, P. Dollár, and R. B. Girshick, “Mask R-CNN,” in Proc. ICCV, 2017, pp. 2900–2908.
[10] Z. Cai and N. Vasconcelos, “Cascade R-CNN: Delving into high quality object detection,” in Proc. CVPR, 2018, pp. 6154–6162.
[11] Z. Liu et al., “Swin transformer: Hierarchical vision transformer using shifted windows,” in Proc. ICCV, 2021, pp. 9992–10002.
[12] Y. Li, Y. Chen, N. Wang, and Z. Zhang, “Scale-aware trident networks for object detection,” in Proc. ICCV, 2019, pp. 6053–6062.
[13] J. Pang, K. Chen, J. Shi, H. Feng, W. Ouyang, and D. Lin, “Libra R-CNN: Towards balanced learning for object detection,” in Proc. CVPR, 2019, pp. 821–830.
[14] X. Zhang, Y. Wei, Y. Yang, and T. S. Huang, “SG-one: Similarity guidance network for one-shot semantic segmentation,” IEEE Trans. Cybern., vol. 50, no. 9, pp. 3855–3865, Sep. 2020.
[15] G. Chen et al., “A survey of the four pillars for small object detection: Multiscale representation, contextual information, super-resolution, and region proposal,” IEEE Trans. Syst., Man, Cybern., Syst., vol. 52, no. 2, pp. 936–953, Feb. 2022.
[16] J. Li, Z. Pan, Q. Liu, Y. Cui, and Y. Sun, “Complementarity-aware attention network for salient object detection,” IEEE Trans. Cybern., vol. 52, no. 2, pp. 873–886, Feb. 2022.
[17] Z. Chen et al., “Hybrid task cascade for instance segmentation,” in Proc. CVPR, 2019, pp. 4969–4978.
[18] K. Sun, B. Xiao, D. Liu, and J. Wang, “Deep high-resolution representation learning for human pose estimation,” in Proc. CVPR, 2019, pp. 5693–5703.
[19] X. Wang, L. Gao, J. Song, and H. T. Shen, “KTN: Knowledge transfer network for multi-person DensePose estimation,” in Proc. ACM/MM, 2020, pp. 3780–3788.
[20] W. Wang, H. Zhu, J. Dai, Y. Pang, J. Shen, and L. Shao, “Hierarchical human parsing with typed part-relation reasoning,” in Proc. CVPR, 2020, pp. 8926–8936.
[21] T. Wang et al., “RecapNet: Action proposal generation mimicking human cognitive process,” IEEE Trans. Cybern., vol. 51, no. 12, pp. 6017–6028, Dec. 2021.
[22] J. Gao, Y. Yuan, and Q. Wang, “Feature-aware adaptation and density alignment for crowd counting in video surveillance,” IEEE Trans. Cybern., vol. 51, no. 10, pp. 4822–4833, Oct. 2021.
[23] K. Gong, X. Liang, Y. Li, Y. Chen, M. Yang, and L. Lin, “Instance-level human parsing via part grouping network,” in Proc. ECCV, 2018, pp. 805–822.
[24] A. Newell, Z. Huang, and J. Deng, “Associative embedding: End-to-end learning for joint detection and grouping,” in Proc. NeurIPS, 2017, pp. 1–11.
[25] Z. Cao, G. Hidalgo, T. Simon, S. Wei, and Y. Sheikh, “OpenPose: Realtime multi-person 2D pose estimation using part affinity fields,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 43, no. 1, pp. 172–186, Jan. 2021.
[26] L. Yang, Q. Song, Z. Wang, and M. Jiang, “Parsing R-CNN for instance-level human analysis,” in Proc. CVPR, 2019, pp. 364–373.
[27] Y. Dai, X. Wang, L. Gao, J. Song, and H. T. Shen, “RSGNet: Relation based skeleton graph network for crowded scenes pose estimation,” in Proc. AAAI, 2021, pp. 1193–1200.
[28] X. Wang, L. Gao, Y. Dai, Y. Zhou, and J. Song, “Semantic-aware transfer with instance-adaptive parsing for crowded scenes pose estimation,” in Proc. ACM/MM, 2021, pp. 686–694.
[29] M. Hao, Y. Liu, X. Zhang, and J. Sun, “LabelfEnc: A new intermediate supervision method for object detection,” in Proc. ECCV, 2020, pp. 529–545.
[30] T. Wang, L. Yuan, X. Zhang, and J. Feng, “Distilling object detectors with fine-grained feature imitation,” in Proc. CVPR, 2019, pp. 4933–4942.
[31] Q. Li, S. Jin, and J. Yan, “Mimicking very efficient network for object detection,” in Proc. CVPR, 2017, pp. 7341–7349.
[32] G. Bertasius, H. Wang, and L. Torresani, “Is space-time attention all you need for video understanding?” in Proc. ICML, 2021, pp. 813–824.
[33] Y. Guo et al., “From general to specific: Informative scene graph generation via balance adjustment,” in Proc. ICCV, 2021, pp. 16363–16372.
[34] R. Zellers, M. Yatskar, S. Thomson, and Y. Choi, “Neural motifs: Scene graph parsing with global context,” in Proc. CVPR, 2018, pp. 5831–5840.
[35] Y. Guo et al., “Relation regularized scene graph generation,” IEEE Trans. Cybern., vol. 52, no. 7, pp. 5961–5972, Jul. 2022.

### Table VIII

| Methods        | Attribute-RCNN | GT-box<sub>part</sub> | 41.9 | 99.9% (±58.0%) | 61.4% (±22.3%) | 66.7% (±22.5%) | 42.1% (±16.1%) | AP<sub>IoU+F<sub>1</sub></sub> |
|----------------|----------------|----------------------|-----|----------------|----------------|----------------|----------------|----------------|

| Methods        | GT-box<sub>person</sub> | GT-box<sub>part</sub> | AP<sub>box<sub>part</sub></sub> | 72.8% | 99.9% (±27.1%) | 86.3% (±1.8%) | 54.6% (±1.1%) | AP<sub>box<sub>part</sub></sub> | Acc<sub>pp</sub> |
|----------------|-------------------------|----------------------|-----------------------------|-----|----------------|----------------|----------------|-----------------|----------------|

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
Xuanhan Wang received the B.E. and M.S. degrees from the University of Electronic Science and Technology of China, Chengdu, China, in 2014 and 2017, respectively, where he is currently pursuing the Ph.D. degree with the School of Computer Science and Engineering. His current research interests include knowledge-driven visual recognition, general human parsing, and human behavior analysis.

Jingkuan Song (Senior Member, IEEE) received the Ph.D. degree from The University of Queensland, Brisbane, QLD, Australia, in 2014. He is currently a Professor with the Shenzhen Institute for Advanced Study, University of Electronic Science and Technology of China, Shenzhen, China. His research interests include large-scale multimedia retrieval, image/video segmentation and image/video understanding using hashing, graph learning, and deep learning techniques.

Lechao Cheng (Member, IEEE) received the Ph.D. degree from the College of Computer Science and Technology, Zhejiang University, Hangzhou, China, in 2019. He is currently an Associate Professor with the Research Center for Intelligent Computing Software, Zhejiang Lab, Hangzhou, China. His research areas center around vision knowledge transfer in deep learning.

Heng Tao Shen (Fellow, IEEE) received the B.Sc. (First Class Hons.) and Ph.D. degrees in computer science from the Department of Computer Science, National University of Singapore, Singapore, in 2000 and 2004, respectively. He is currently a Professor, the Dean of the School of Computer Science and Engineering, the Executive Dean of the Artificial Intelligence Research Institute, and the Director of the Center for Future Media, University of Electronic Science and Technology of China, Chengdu, China. He has published over 250 peer-reviewed papers. His current research interests include multimedia search, computer vision, artificial intelligence, and big data management. Prof. Shen received seven best paper awards from international conferences, including the Best Paper Award from ACM Multimedia 2017 and the Best Paper Award-Honorable Mention from ACM SIGIR 2017. He has served as a General Co-Chair for ACM Multimedia 2021 and the Program Committee Co-Chair for ACM Multimedia 2015, and is an Associate Editor of the ACM Transactions of Data Science, IEEE TRANSACTIONS ON IMAGE PROCESSING, IEEE TRANSACTIONS ON MULTIMEDIA, and IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING.