



Multi-teacher Knowledge Distillation for End-to-End Text Image Machine Translation

Cong Ma^{1,2}, Yaping Zhang^{1,2(✉)}, Mei Tu⁴, Yang Zhao^{1,2}, Yu Zhou^{2,3},
and Chengqing Zong^{1,2}

- ¹ School of Artificial Intelligence, University of Chinese Academy of Sciences,
Beijing 100049, People's Republic of China
{cong.ma,yaping.zhang,yang.zhao,cqzong}@nlpr.ia.ac.cn
- ² State Key Laboratory of Multimodal Artificial Intelligence Systems (MAIS),
Institute of Automation, Chinese Academy of Sciences, Beijing 100190,
People's Republic of China
yzhou@nlpr.ia.ac.cn
- ³ Fanyu AI Laboratory, Zhongke Fanyu Technology Co., Ltd., Beijing 100190,
People's Republic of China
- ⁴ Samsung Research China - Beijing (SRC-B), Beijing, China
mei.tu@samsung.com

Abstract. Text image machine translation (TIMT) has been widely used in various real-world applications, which translates source language texts in images into another target language sentence. Existing methods on TIMT are mainly divided into two categories: the recognition-then-translation pipeline model and the end-to-end model. However, how to transfer knowledge from the pipeline model into the end-to-end model remains an unsolved problem. In this paper, we propose a novel Multi-Teacher Knowledge Distillation (MTKD) method to effectively distillate knowledge into the end-to-end TIMT model from the pipeline model. Specifically, three teachers are utilized to improve the performance of the end-to-end TIMT model. The image encoder in the end-to-end TIMT model is optimized with the knowledge distillation guidance from the recognition teacher encoder, while the sequential encoder and decoder are improved by transferring knowledge from the translation sequential and decoder teacher models. Furthermore, both token and sentence-level knowledge distillations are incorporated to better boost the translation performance. Extensive experimental results show that our proposed MTKD effectively improves the text image translation performance and outperforms existing end-to-end and pipeline models with fewer parameters and less decoding time, illustrating that MTKD can take advantage of both pipeline and end-to-end models. Our codes are available at: https://github.com/EriCongMa/MTKD_TIMT.

Keywords: Text Image Machine Translation · Knowledge Distillation · Machine Translation

1 Introduction

Text image machine translation (TIMT) is a cross-modal generation task, which translates source language texts in images into target language sentences. Various real-world applications have been conducted for TIMT, such as digital document translation, scene text translation, handwritten text image translation, and so on. Existing TIMT systems are mainly constructed with a recognition-then-translation pipeline model [1, 4, 7, 9, 15], which first recognizes texts in images by a text image recognition (TIR) model [2, 16, 17, 27, 28], and then generates target language translation with a machine translation (MT) model [20, 22, 29, 30]. However, pipeline models have to train and deploy two separate models, leading to parameter redundancy and slow decoding speed. Meanwhile, errors in TIR model are further propagated by MT models, which causes more translation mistakes in the final translation results.

To address the shortcomings of pipeline models, end-to-end TIMT models are proposed with a more efficient architecture [14]. Although end-to-end models have fewer parameters and faster decoding speed, the end-to-end training data is limited compared with recognition or translation datasets, leading to inadequate training and limited translation performance of end-to-end models. As a result, how to explicitly incorporate external recognition or translation results has been studied by existing research [6, 13]. Furthermore, transfer knowledge from TIR or MT models has been conducted to end-to-end TIMT models through feature transformation [18] and cross-modal mimic framework [5].

However, sub-modules in end-to-end TIMT models play quite different functions, which need different knowledge from various teacher models. Although existing methods explore to transfer knowledge from external models, how to introduce different knowledge into each sub-modules of the end-to-end TIMT model remains unsolved.

In this paper, we propose a novel multi-teacher knowledge distillation (MTKD) approach for end-to-end TIMT model, which is designed to transfer various types of knowledge into end-to-end TIMT model. Specifically, three sub-modules in end-to-end models are considered to optimize by distilling knowledge from different teacher models.

- Image encoder aims at extracting features of input images from pixel space to dense feature space, which has a similar function as the TIR image encoder. As a result, TIR image encoder is utilized as the teacher model for image encoder in end-to-end TIMT model to improve the image feature extraction.
- Sequential encoder in end-to-end TIMT model fuses the local image features into contextual features, which learns advanced semantic information of the sentences in text images. To guide semantic feature learning, MT sequential encoder offers the teacher guidance for TIMT sequential encoder to better map image features into semantic features.
- Decoder in end-to-end TIMT model generates target translation autoregressively, which has a similar function as the MT decoder. As so, the prediction distribution on target language vocabulary is utilized as the teacher distri-

bution to guide the decoder in end-to-end TIMT generate better prediction distribution.

By transferring different knowledge into corresponding sub-modules in end-to-end TIMT model, fine-grained knowledge distillation can better improve the translation quality of end-to-end TIMT models. In summary, our contributions are summarized as:

- We propose a novel multi-teacher knowledge distillation method for end-to-end TIMT model, which is carefully designed for fine-grained knowledge transferring to various sub-modules in end-to-end TIMT models.
- Various teacher knowledge distillation provides more improvements compared with single teacher guidance, indicating different sub-modules in end-to-end models need different knowledge information to better adapt corresponding functions.
- Extensive experimental results show our proposed MTKD method can effectively improve the translation performance of end-to-end TIMT models. Furthermore, MTKD based TIMT model also outperforms pipeline system with fewer parameters and less decoding time.

2 Related Work

2.1 Text Image Machine Translation

Text image machine translation models are mainly divided into pipeline and end-to-end models. Pipeline models deploy text image recognition and machine translation models respectively. Specifically, the source language text images are first fed into TIR models to obtain the recognized source language sentences. Second, the source language sentences are translated into the target language with the MT model. Various applications have been conducted with the pipeline TIMT architectures. Photos, scene images, document images, and manga pages are taken as the input text images. The TIR model recognizes the source language texts, and the MT model generates target language translation [1, 3, 4, 7, 9, 23, 25, 26].

End-to-end TIMT models face the problem of end-to-end data scarcity and the performance is limited. To address the problem of data limitation, a multi-task learning method is proposed to incorporate external datasets [6, 13, 18]. Feature transformation module is proposed to bridge pre-trained TIR encoder and MT decoder [18]. The hierarchy Cross-Modal Mimic method is proposed to utilize MT model as a teacher model to guide the end-to-end TIMT student model [5].

2.2 Knowledge Distillation

Knowledge distillation has been widely used to distillate external knowledge into the student model to improve performance, speed up the training process, and

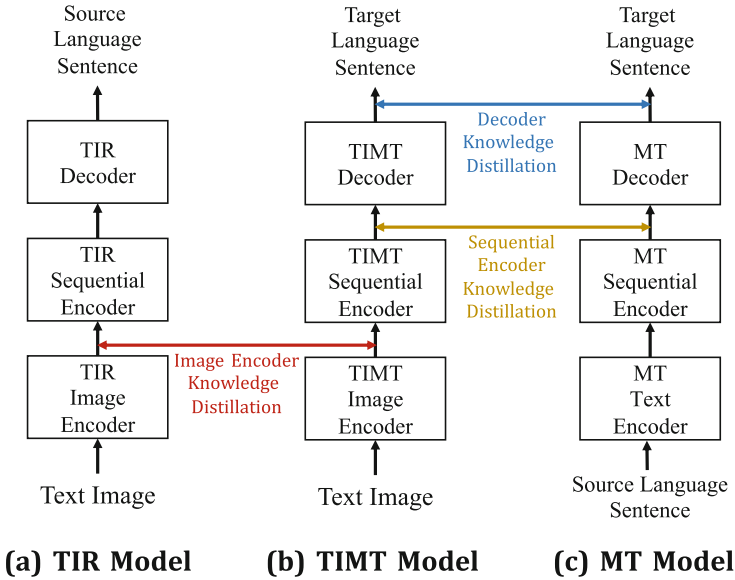


Fig. 1. Overall Diagram of (a) Text Image Recognition, (b) Text Image Machine Translation, (c) Machine Translation models and Multi-Teacher Knowledge Distillation.

decrease the parameter amounts in teacher models. Specifically, in sequence-to-sequence generation related tasks, token-level and sentence-level knowledge distillation have been proven effective in generation tasks [10, 11]. Various tasks have been significantly improved through knowledge distillation method, like bilingual neural machine translation [19], multi-lingual translation [21], and speech translation [12].

To incorporate more knowledge into one student model, multiple teacher models are utilized in some studies to further transfer knowledge into student model. [21] proposed to use various teacher models in different training mini-batch to make the multilingual NMT model learn various language knowledge. DOPE is designed to incorporate multiple teacher models to guide different subnetworks of the student model to provide fine-grained knowledge like body, hand, and face segmentation information [24].

However, existing methods lack exploration in integrating various knowledge into end-to-end TIMT models. Our proposed multi-teacher knowledge distillation effectively addresses this problem by transferring different knowledge into various sub-modules to meet the corresponding functional characteristics of different modules.

3 Methodology

3.1 Problem Definition

The end-to-end TIMT model aims at translating source language texts in images into target language sentences. Let \mathbf{I} be the source language text image and corresponding target language sentence is \mathbf{Y} containing z tokens $\{y^1, y^2, \dots, y^z\}$. The training object for the end-to-end TIMT model is to maximize the translation probability:

$$P(\mathbf{Y}|\mathbf{I}; \theta_{\text{TIMT}}) = \prod_{i=1}^z P(y^i | \mathbf{I}, \mathbf{Y}_{<i}) \quad (1)$$

where $\mathbf{Y}_{<i}$ represents the translation history at the i -th decoding step, and θ_{TIMT} denotes the parameters of end-to-end TIMT model.

Specifically, to generate target language translation, end-to-end TIMT model is divided into three sub-modules: image encoder, sequential encoder, and decoder as shown in Fig. 1 (b). Image encoder \mathcal{I} extracts image features from pixel space and ResNet [8] is utilized as the image encoder in our work:

$$F_{\mathcal{I}} = \mathcal{I}(\mathbf{I}; \theta_{\mathcal{I}}) = \text{ResNet}(\mathbf{I}) \quad (2)$$

where $\mathbf{I} \in \mathbb{R}^{H \cdot W \cdot C}$ denotes the input text image, and H, W, C represent the height, width, and channel of input image respectively. $F_{\mathcal{I}} \in \mathbb{R}^{l_{\mathcal{I}} \cdot c}$ denotes the image feature, and $l_{\mathcal{I}}, c$ represent length and channel of feature sequence respectively. Generally, image features encoded by convolutional network are $F'_{\mathcal{I}} \in \mathbb{R}^{h \cdot w \cdot c}$, where h, w, c represent the height, width, and channel of feature maps respectively. To meet the requirement of following sequential encoding, feature maps are resized to feature sequence by reducing height and width dimension into feature length: $l_{\mathcal{I}} = h \cdot w$. Thus, the output of image encoder is a feature sequence containing local information of input text image.

Sequential encoder $\mathcal{S}(\cdot)$ aims at encoding contextual semantic features given local features of input text image. Transformer encoder is utilized as the sequential encoder in this paper:

$$F_{\mathcal{S}} = \mathcal{S}(F_{\mathcal{I}}; \theta_{\mathcal{S}}) = \text{TransformerEncoder}(F_{\mathcal{I}}) \quad (3)$$

where $F_{\mathcal{S}} \in \mathbb{R}^{l_{\mathcal{S}} \cdot h_{\mathcal{S}}}$ represents the sequential features that contains contextual semantic information of the whole feature sequence. $l_{\mathcal{S}}, h_{\mathcal{S}}$ represent sequence length and hidden dimension of sequential features.

Finally, target language decoder $\mathcal{D}(\cdot)$ generates translation results autoregressively and transformer decoder is utilized in our work:

$$F_{\mathcal{D}} = \mathcal{D}(F_{\mathcal{S}}; \theta_{\mathcal{D}}) = \text{TransformerDecoder}(F_{\mathcal{S}}) \quad (4)$$

where $F_{\mathcal{D}} \in \mathbb{R}^{l_{\mathcal{D}} \cdot h_{\mathcal{D}}}$ represents the output of decoder. $l_{\mathcal{D}}, h_{\mathcal{D}}$ represent sequence length and hidden dimension of decoder features respectively. The final decoded word \hat{y}_{TIMT}^i is calculated by:

$$\hat{y}_{\text{TIMT}}^i = \arg \max_{j \in [1, |V_{\mathbf{Y}}|]} P(\hat{y}_j^i | \mathbf{I}, \hat{\mathbf{Y}}_{<i}), \text{ where } P(\hat{y}_j^i | \mathbf{I}, \hat{\mathbf{Y}}_{<i}) \propto W_o F_{\mathcal{D}}^i \quad (5)$$

where $P(\hat{y}_j^i | \mathbf{I}, \hat{\mathbf{Y}}_{<i})$ denotes the probability that the decoder predicts the j -th word \hat{y}_j^i in vocabulary at i -th decoding step. $W_o \in \mathbb{R}^{|V_{\mathbf{Y}}| \times h_{\mathcal{D}}}$ denotes a linear matrix that maps decoder features into target language words. $|V_{\mathbf{Y}}|, h_{\mathcal{D}}$ represent the size of target language vocabulary and the hidden dimension of decoder respectively. $F_{\mathcal{D}}^i$ means the i -th element of decoder feature $F_{\mathcal{D}}$, which represents the decoder information at position i . $\hat{\mathbf{Y}}_{<i}$ represents the translation history before i -th step. In summary, end-to-end TIMT model utilizes image encoder, sequential encoder, and target language decoder to generate target language translation results word by word.

To optimize the end-to-end TIMT model, the log-likelihood loss function is utilized:

$$\begin{aligned} \mathcal{L}_{\text{TIMT}} &= - \sum_{(\mathbf{I}, \mathbf{Y}) \in \mathbf{D}_{\text{TIMT}}} \log P(\mathbf{Y} | \mathbf{I}) \\ \log P(\mathbf{Y} | \mathbf{I}) &= \sum_i^z \sum_j^{|V_{\mathbf{Y}}|} \mathbb{I}(\hat{y}_j^i = y^i) \log P(\hat{y}_j^i | \mathbf{I}, \hat{\mathbf{Y}}_{<i}) \end{aligned} \quad (6)$$

where $\mathbb{I}(\hat{y}_j^i = y^i)$ is an indicator function which equals 1 when predicted word \hat{y}_j^i is the same as the ground-truth y^i , otherwise it equals 0. z denotes the sentence length of target language ground-truth. \mathbf{D}_{TIMT} represents the text image translation training dataset.

3.2 Architecture of Teacher Models

Different sub-modules in end-to-end TIMT model play quite different functions and need various knowledge guidance. Image encoder is utilized to extract local visual features from input text images, while a sequential encoder further encodes contextual semantic information from local visual features. Finally, a decoder is designed to generate translation results given sequential features. To incorporate various knowledge into sub-modules of end-to-end TIMT model, three teacher models are utilized to guide the optimization of image encoder, sequential encoder, and decoder respectively. Specifically, knowledge of extracting text image features is transferred from TIR encoder. MT sequential encoder provides the guidance of contextual semantic feature learning, while MT decoder distillates the target language generation knowledge into TIMT decoder.

Text Image Recognition Teacher Model. Considering image encoder extracts local visual features from input text images, which is consistent between TIMT and TIR tasks, TIR model is incorporated to provide guidance for image feature learning. In this paper, TIR models are also divided into three sub-modules as end-to-end TIMT model to better understand the information flow

between teacher and student models. Similar to TIMT image encoder, TIR image encoder also aims at extracting local visual features of input text images:

$$F_{\mathcal{I}}^{\text{TIR}} = \mathcal{I}^{\text{TIR}}(\mathbf{I}; \theta_{\mathcal{I}}^{\text{TIR}}) = \text{ResNet}(\mathbf{I}) \quad (7)$$

where $F_{\mathcal{I}}^{\text{TIR}}$ denotes the image features encoded by TIR image encoder $\mathcal{I}^{\text{TIR}}(\cdot)$ and the dimension of $F_{\mathcal{I}}^{\text{TIR}}$ is same as the image feature $F_{\mathcal{I}}$ of end-to-end TIMT model introduced in Sect. 3.1. $\theta_{\mathcal{I}}^{\text{TIR}}$ represents the model parameters of TIR image encoder. The architecture of TIR image encoder is similar to TIMT image encoder, but these two models are trained with different supervised data.

TIR Sequential encoder is also designed to further extract contextual information by considering whole local visual features:

$$F_{\mathcal{S}}^{\text{TIR}} = \mathcal{S}^{\text{TIR}}(F_{\mathcal{I}}^{\text{TIR}}; \theta_{\mathcal{S}}^{\text{TIR}}) = \text{TransformerEncoder}(F_{\mathcal{I}}^{\text{TIR}}) \quad (8)$$

where $F_{\mathcal{S}}^{\text{TIR}}$, $\mathcal{S}^{\text{TIR}}(\cdot)$, $\theta_{\mathcal{S}}^{\text{TIR}}$ denote TIR sequential features, TIR sequential encoder, and parameters of TIR sequential encoder respectively.

Different from generating target language in TIMT decoder, TIR decoder predicts source language words autoregressively:

$$F_{\mathcal{D}}^{\text{TIR}} = \mathcal{D}^{\text{TIR}}(F_{\mathcal{S}}^{\text{TIR}}; \theta_{\mathcal{D}}^{\text{TIR}}) = \text{TransformerDecoder}(F_{\mathcal{S}}^{\text{TIR}}) \quad (9)$$

where $F_{\mathcal{D}}^{\text{TIR}}$, $\mathcal{D}^{\text{TIR}}(\cdot)$, $\theta_{\mathcal{D}}^{\text{TIR}}$ denote TIR decoder features, TIR decoder, and parameters of TIR decoder respectively. To further map TIR decoder feature into source language space, a transformation matrix is utilized to transform decoder feature into source language word:

$$\hat{x}_{\text{TIR}}^i = \arg \max_{j \in [1, |\mathbf{V}_{\mathbf{X}}|]} P(\hat{x}_j^i | \mathbf{I}, \hat{\mathbf{X}}_{<i}), \quad \text{where } P(\hat{x}_j^i | \mathbf{I}, \hat{\mathbf{X}}_{<i}) \propto W_o^{\text{TIR}} F_{\mathcal{D}}^{\text{TIR}^i} \quad (10)$$

where \hat{x}_j^i represents the j -th word in source language vocabulary at decoding position i , while \hat{x}_{TIR}^i represents the final predicted word of decoder at i -th decoding step. $W_o^{\text{TIR}} \in \mathbb{R}^{|\mathbf{V}_{\mathbf{X}}| \cdot h_{\mathcal{D}}^{\text{TIR}}}$ denotes the transformation matrix from decoder feature space to source language space. $|\mathbf{V}_{\mathbf{X}}|$, $h_{\mathcal{D}}^{\text{TIR}}$ represent the size of source language vocabulary and feature dimension of TIR decoder respectively. $F_{\mathcal{D}}^{\text{TIR}^i}$ denotes the TIR decoder feature at position i . $\hat{\mathbf{X}}_{<i}$ represents the recognition history before i -th decoding step.

The overall architecture of TIR and TIMT models is similar, but the supervised data is different. TIR model is trained with recognition data pair $\langle \mathbf{I}, \mathbf{X} \rangle$, where \mathbf{X} means the source language recognition label of input text image \mathbf{I} . While TIMT model is trained with text image translation pair $\langle \mathbf{I}, \mathbf{Y} \rangle$, where \mathbf{Y} means the target language translation of corresponding source language sentence \mathbf{X} . To optimize the parameters in TIR model, the log-likelihood loss is utilized similar to TIMT optimization:

$$\begin{aligned} \mathcal{L}_{\text{TIR}} &= - \sum_{(\mathbf{I}, \mathbf{X}) \in \mathbf{D}_{\text{TIR}}} \log P(\mathbf{X} | \mathbf{I}) \\ \log P(\mathbf{X} | \mathbf{I}) &= \sum_i \sum_j^{|\mathbf{V}_{\mathbf{X}}|} \mathbb{I}(\hat{x}_j^i = x^i) \log P(\hat{x}_j^i | \mathbf{I}, \hat{\mathbf{X}}_{<i}) \end{aligned} \quad (11)$$

where \hat{x}_j^i denotes the j -th word in source language vocabulary at i -th decoding step, while x^i represents the ground-truth word at i -th decoding step. z denotes the sentence length of ground-truth. $\mathbb{I}(\cdot)$ means the indicator function as introduced in Eq. (6). \mathbf{D}_{TIR} represents the text image recognition dataset.

Machine Translation Teacher Model. Different from cross-modal generation TIR and TIMT models, MT model is a text-to-text transformation network. Thus, the encoder of raw data is quite different from TIR and TIMT models. To obtain text features from source language sentence strings, an embedding layer based text encoder is utilized to map the input words into word embedding:

$$F_{\mathcal{T}}^{\text{MT}} = \mathcal{T}^{\text{MT}}(\mathbf{X}; \theta_{\mathcal{T}}^{\text{MT}}) = \text{Embedding}(\mathbf{X}) \quad (12)$$

where $F_{\mathcal{T}}^{\text{MT}}$, $\mathcal{T}^{\text{MT}}(\cdot)$, $\theta_{\mathcal{T}}^{\text{MT}}$ represent text features, MT text encoder, and parameters of MT text encoders respectively.

Word embedding only contains single word information rather than global semantic information. To better extract contextual semantic features, MT sequential encoder further encodes contextual information by considering all input words:

$$F_{\mathcal{S}}^{\text{MT}} = \mathcal{S}^{\text{MT}}(F_{\mathcal{T}}^{\text{MT}}; \theta_{\mathcal{S}}^{\text{MT}}) = \text{TransformerEncoder}(F_{\mathcal{T}}^{\text{MT}}) \quad (13)$$

where $F_{\mathcal{S}}^{\text{MT}}$, $\mathcal{S}^{\text{MT}}(\cdot)$, $\theta_{\mathcal{S}}^{\text{MT}}$ denote MT sequential feature, MT sequential encoder, and parameters of MT sequential encoder respectively. Similar to TIR and TIMT sequential encoder, transformer encoder is utilized to extract contextual semantic features given MT text features.

MT decoder generates target language translation word by word given MT sequential features:

$$F_{\mathcal{D}}^{\text{MT}} = \mathcal{D}^{\text{MT}}(F_{\mathcal{S}}^{\text{MT}}; \theta_{\mathcal{D}}^{\text{MT}}) = \text{TransformerDecoder}(F_{\mathcal{S}}^{\text{MT}}) \quad (14)$$

where $F_{\mathcal{D}}^{\text{MT}}$, $\mathcal{D}^{\text{MT}}(\cdot)$, $\theta_{\mathcal{D}}^{\text{MT}}$ represent MT decoder features, MT decoder, and parameters of MT decoder respectively. To further map MT decoder features into target language space, a transformation matrix is utilized to calculate the translation probability:

$$\hat{y}_{\text{MT}}^i = \arg \max_{j \in [1, |V_{\mathbf{Y}}|]} P(\hat{y}_j^i | \mathbf{X}, \hat{\mathbf{Y}}_{<i}), \text{ where } P(\hat{y}_j^i | \mathbf{X}, \hat{\mathbf{Y}}_{<i}) \propto W_o^{\text{MT}} F_{\mathcal{D}}^{\text{MT}^i} \quad (15)$$

where \hat{y}_j^i represents the j -th word in target language vocabulary at i -th decoding step, while \hat{y}_{MT}^i represents the final predicted word of target language decoder at decoding position i . \mathbf{X} , $\hat{\mathbf{Y}}_{<i}$ denote source language sentence and translation history before i -th decoding step respectively. $W_o^{\text{MT}} \in \mathbb{R}^{|V_{\mathbf{Y}}| \cdot h_{\mathcal{D}}^{\text{MT}}}$ denotes the transformation matrix which maps MT decoder features into target language space. $|V_{\mathbf{Y}}|$, $h_{\mathcal{D}}^{\text{MT}}$ denote the size of target language vocabulary and hidden dimension of MT decoder feature respectively.

$$\begin{aligned} \mathcal{L}_{\text{MT}} &= - \sum_{(\mathbf{X}, \mathbf{Y}) \in \mathbf{D}_{\text{MT}}} \log P(\hat{\mathbf{Y}}|\mathbf{X}) \\ \log P(\hat{\mathbf{Y}}|\mathbf{X}) &= \sum_i^z \sum_j^{|V_{\mathbf{Y}}|} \mathbb{I}(\hat{y}_j^i = y^i) \log P(\hat{y}_j^i|\mathbf{X}, \hat{\mathbf{Y}}_{<i}) \end{aligned} \quad (16)$$

where \hat{y}_j^i denotes the j -th word in target language vocabulary at i -th decoding step, while y^i represents ground-truth word at i -th decoding step. z denotes the sentence length of ground-truth. $\mathbb{I}(\cdot)$ means the indicator function as introduced in Eq. (6). \mathbf{D}_{MT} represents the text machine translation dataset.

From the comparison of TIR, MT, and TIMT architectures, they have similar and different functions. For example, TIR image encoder and TIMT image encoder have similar structure and functions. All the sequential encoders are similar in architecture and the functions all aim at extracting contextual semantic information. Furthermore, MT decoder and TIMT decoder are both designed to predict target language sentences, which has similar structure and function. As a result, sub-modules of TIMT model with similar architecture and function can as that of TIR or MT models can be improved by multi-teacher knowledge distillation.

3.3 Knowledge Distillation from TIR Image Encoder

TIMT image encoder and TIR image encoder both extract local visual features from input text images. Compared with TIMT task, TIR task has much more training data, thus TIR models can be better optimized to encode image features of text images. To address the data limitation of end-to-end TIMT task, knowledge distillation from TIR image encoder is proposed to transfer text image encoding knowledge into TIMT image encoder. As shown in Fig. 1, TIMT image encoder is optimized not only by end-to-end text image translation loss but also by the guidance from TIR image encoder. To align the TIMT image features with TIR image features, both token-level and sentence-level knowledge distillation are incorporated to guide TIMT image encoder to predict similar image features as TIR image features:

Token-Level Image Encoder Knowledge Distillation. TIMT and TIR image features are feature sequences as introduced in Sect. 3.1. To provide fine-grained guidance information, L2-Norm constraint is utilized to guide TIMT image encoder outputs:

$$\mathcal{L}_{\text{TKD}}^{\mathcal{I}} = \frac{1}{B \cdot l_{\mathcal{I}}} \sum_j^B \sum_i^{l_{\mathcal{I}}} \|F_{\mathcal{I}}^{ij} - F_{\mathcal{I}}^{\text{TIR}^{ij}}\|_2 \quad (17)$$

where $\mathcal{L}_{\text{TKD}}^{\mathcal{I}}$ denotes the token-level image encoder knowledge distillation loss function. $F_{\mathcal{I}}^{ij}, F_{\mathcal{I}}^{\text{TIR}^{ij}}$ represent TIMT and TIR image features of j -th sample at

position i respectively. $l_{\mathcal{I}}$ denotes the length of TIMT image feature sequence, and $l_{\mathcal{I}} = l_{\mathcal{I}}^{\text{TIR}}$ in our experiments, indicating the sequence length of TIMT and TIR image features are the same. B denotes the batch size.

Sentence-Level Image Encoder Knowledge Distillation. To provide sentence-level guidance, both TIMT and TIR global image features are calculated by average pooling:

$$\mathcal{L}_{\text{SKD}}^{\mathcal{I}} = \frac{1}{B} \sum_j \left\| \frac{1}{l_{\mathcal{I}}} \sum_i^{l_{\mathcal{I}}} F_{\mathcal{I}}^{ij} - \frac{1}{l_{\mathcal{I}}^{\text{TIR}}} \sum_i^{l_{\mathcal{I}}^{\text{TIR}}} F_{\mathcal{I}}^{\text{TIR}ij} \right\|_2 \quad (18)$$

where $\mathcal{L}_{\text{SKD}}^{\mathcal{I}}$ represents the loss function of sentence-level image encoder knowledge distillation. By calculating the global image features, the optimization of TIMT image encoder is guided by the global alignment between TIMT and TIR image features.

Finally, the token-level and sentence-level image encoder knowledge distillation loss functions are fused to obtain image encoder knowledge distillation loss function $\mathcal{L}_{\text{KD}}^{\mathcal{I}}$, which provides multi-granularity knowledge distillation guidance information:

$$\mathcal{L}_{\text{KD}}^{\mathcal{I}} = \mathcal{L}_{\text{TKD}}^{\mathcal{I}} + \mathcal{L}_{\text{SKD}}^{\mathcal{I}} \quad (19)$$

3.4 Knowledge Distillation from MT Sequential Encoder

The sequential encoder is vital to TIMT task, because the contextual semantic features are important for cross-lingual generation. To improve the ability of TIMT sequential encoder, knowledge distillation from MT sequential encoder is incorporated to guide the optimization of TIMT sequential encoder as shown in Fig. 1. Similar to image encoder knowledge distillation, sequential encoder knowledge distillation also has token-level and sentence-level knowledge distillations:

Token-Level Sequential Encoder Knowledge Distillation. Similar to the token-level image encoder knowledge distillation, MT sequential features are regarded as the guidance for TIMT sequential features through L2-Norm constraint:

$$\mathcal{L}_{\text{TKD}}^{\mathcal{S}} = \frac{1}{B \cdot l_{\mathcal{S}}} \sum_j \sum_i^{l_{\mathcal{S}}} \|F_{\mathcal{S}}^{ij} - F_{\mathcal{S}}^{\text{MT}ij}\|_2 \quad (20)$$

where $\mathcal{L}_{\text{TKD}}^{\mathcal{S}}$ represents sequential knowledge distillation loss function. $F_{\mathcal{S}}^{ij}, F_{\mathcal{S}}^{\text{MT}ij}$ represent TIMT and MT sequential features of j -th sample at position i respectively. $l_{\mathcal{S}}$ denotes the length of TIMT sequential feature sequence, which is set the same as the length of MT sequential feature sequence $l_{\mathcal{S}}^{\text{MT}}$.

Sentence-Level Sequential Encoder Knowledge Distillation. To further provide global guidance of sequential feature learning, the sentence-level sequential encoder knowledge distillation is proposed by performing average pooling on TIMT and MT sequential features:

$$\mathcal{L}_{\text{SKD}}^S = \frac{1}{B} \sum_j \left\| \frac{1}{l_S} \sum_i F_S^{ij} - \frac{1}{l_S^{\text{MT}}} \sum_i F_S^{\text{MT}^{ij}} \right\|_2 \quad (21)$$

where $\mathcal{L}_{\text{SKD}}^S$ denotes the sequential encoder knowledge distillation loss function. The length of TIMT and MT sequential features are the same ($l_S = l_S^{\text{MT}}$) as introduced in token-level sequential encoder knowledge distillation.

Overall sequential encoder knowledge distillation loss function $\mathcal{L}_{\text{KD}}^S$ is obtained by combining token-level and sentence-level sequential encoder knowledge distillation:

$$\mathcal{L}_{\text{KD}}^S = \mathcal{L}_{\text{TKD}}^S + \mathcal{L}_{\text{SKD}}^S \quad (22)$$

3.5 Knowledge Distillation from MT Decoder

Different from image and sequential encoder knowledge distillation, decoder knowledge distillation is proposed to align the predicted target language vocabulary distribution between TIMT and MT decoders. Token-level decoder knowledge distillation aims at aligning the prediction probability between TIMT and MT decoders at each decoding step, while sentence-level decoder knowledge distillation takes the MT predicted target language sentence as the ground-truth to calculate the decoding loss for the optimization of TIMT model.

Token-Level Decoder Knowledge Distillation. As introduced in Eq. (5), TIMT decoder predicts the j -th target language word at i -th decoding step with the probability of $P(\hat{y}_j^i | \mathbf{I}, \hat{\mathbf{Y}}_{<i}^{\text{TIMT}})$, while MT decoder generates the j -th target language word at i -th step with the probability of $P(\hat{y}_j^i | \mathbf{X}, \hat{\mathbf{Y}}_{<i}^{\text{MT}})$ as in Eq. (15). To align the decoding distribution, \mathbf{I} and \mathbf{X} are paired text images and corresponding source language text sentences. $\hat{\mathbf{Y}}_{<i}^{\text{TIMT}}$, $\hat{\mathbf{Y}}_{<i}^{\text{MT}}$ represent decoding history of TIMT and MT models respectively. The token-level decoder knowledge distillation loss is calculated by updating the vanilla cross-entropy loss:

$$\mathcal{L}_{\text{TKD}}^D = - \sum_i \sum_j^{|\mathcal{V}_Y|} P(\hat{y}_j^i | \mathbf{X}, \hat{\mathbf{Y}}_{<i}^{\text{MT}}) \log P(\hat{y}_j^i | \mathbf{I}, \hat{\mathbf{Y}}_{<i}^{\text{TIMT}}) \quad (23)$$

where $\mathcal{L}_{\text{TKD}}^D$ denotes the token-level decoder knowledge distillation loss. By transferring decoding knowledge from MT teacher decoder, the TIMT decoder is guided to have a similar predicted probability of target language words.

Sentence-Level Decoder Knowledge Distillation. To provide sentence-level decoding knowledge distillation, the MT model decoded target language sentences are utilized to replace original ground-truth sentences. Different from token-level decoder knowledge distillation, which is designed to align the decoding probability between TIMT and MT decoders, sentence-level decoder knowledge distillation aims at guiding the TIMT decoder to have similar translation results as MT decoder:

$$\mathcal{L}_{\text{SKD}}^{\mathcal{D}} = - \sum_i^z \sum_j^{|V_{\mathbf{Y}}|} \mathbb{I}(\hat{y}_j^i = \hat{y}_{\text{MT}}^i) \log P(\hat{y}_j^i | \mathbf{I}, \hat{\mathbf{Y}}_{<i}^{\text{TIMT}}) \quad (24)$$

where $\mathcal{L}_{\text{SKD}}^{\mathcal{D}}$ denotes sequence-level decoder knowledge distillation loss function. Different from the vanilla log-likelihood loss function, the ground-truth sentence is replaced as the MT prediction results. Thus the indicator function $\mathbb{I}(\hat{y}_j^i = \hat{y}_{\text{MT}}^i)$ equals 1 when the TIMT decoded word \hat{y}_j^i is the same as the MT predicted word \hat{y}_{MT}^i . By incorporating both token-level and sentence-level decoder knowledge distillation, the overall loss function of decoder knowledge distillation is formulated as:

$$\mathcal{L}_{\text{KD}}^{\mathcal{D}} = \mathcal{L}_{\text{TKD}}^{\mathcal{D}} + \mathcal{L}_{\text{SKD}}^{\mathcal{D}} \quad (25)$$

The final loss function is the combination of end-to-end text image translation and knowledge distillation loss functions:

$$\begin{aligned} \mathcal{L}_{\text{ALL}} &= (1 - \lambda_{\text{KD}}) \mathcal{L}_{\text{TIMT}} + \lambda_{\text{KD}} \mathcal{L}_{\text{KD}} \\ \mathcal{L}_{\text{KD}} &= \lambda_{\mathcal{I}} \mathcal{L}_{\text{KD}}^{\mathcal{I}} + \lambda_{\mathcal{S}} \mathcal{L}_{\text{KD}}^{\mathcal{S}} + \lambda_{\mathcal{D}} \mathcal{L}_{\text{KD}}^{\mathcal{D}} \end{aligned} \quad (26)$$

where λ_{KD} , $\lambda_{\mathcal{I}}$, $\lambda_{\mathcal{S}}$, $\lambda_{\mathcal{D}}$ represent the loss weight of overall knowledge distillation, image encoder knowledge distillation, sequential encoder knowledge distillation, and decoder knowledge distillation respectively.

Source Language Text Images	
Source Language Text	“Use of our PCs in the seminar room with the Internet access”
Target Language Text	“Benutzung unserer PCs im Seminarraum LAN-Zugang”
(a) Example of English-to-German synthetic text image machine translation data set.	
Source Language Text Images	
Source Language Text	“there is one rule we do not break!”
Target Language Text	“我们这里有一条雷打不动的规则!”
(b) Example of English-to-Chinese subtitle text image machine translation data set.	
Source Language Text Images	
Source Language Text	“前方路况繁杂”
Target Language Text	“The road ahead is very complicated”
(c) Example of Chinese-to-English street-view text image machine translation data set.	

Fig. 2. Examples of synthetic, subtitle and street-view text image translation datasets.

4 Experiments

4.1 Datasets

To train the end-to-end TIMT model, the publicly available dataset released by [13] is utilized in our experiments. As shown in Fig. 2, this dataset contains samples from three domains: synthetic, subtitle, and street-view domains. The training and validation samples are all from synthetic domain, while samples in evaluation set are from all three domains. Three translation directions are conducted in this dataset: English-to-Chinese (EnCh), English-to-German (EnDe), and Chinese-to-English (ChEn) translation. There are 1,000,000 training samples, 2,000 validation samples, and 2,000 evaluation samples in synthetic domain. The subtitle test set contains 1,040 samples, while the street-view test set has 1,198 samples. To implement knowledge distillation, triple-aligned samples **{source language images, source language texts, target language texts}** are utilized to transfer the pre-trained knowledge from TIR and MT teacher models into the TIMT student model.

Table 1. Results of various knowledge distillation combinations on English-to-Chinese translation validation set. TKD and SKD represent using single token-level or sentence-level knowledge distillation loss. TKD+SKD means the fused token-level and sentence-level knowledge distillation are used for knowledge distillation loss function. BLEU Score is utilized to evaluate the translation performance.

No	$\lambda_{\mathcal{I}}$	$\lambda_{\mathcal{S}}$	$\lambda_{\mathcal{D}}$	TKD	SKD	TKD+SKD
1	0	0	1	23.02	22.68	23.16
2	0	1	0	22.63	22.44	22.85
3	0	1	1	23.47	23.04	23.79
4	1	0	0	22.45	22.30	22.68
5	1	0	1	23.28	22.95	23.52
6	1	1	0	23.19	22.73	23.34
7	1	1	1	23.86	23.51	24.13

4.2 Experimental Setup

To provide a fair comparison with existing research on end-to-end TIMT task, a similar model architecture as [13] is utilized in our experiment. The TIMT image encoder is composed of TPS Net and Res Net, which extracts the image features from the raw input text images. The TIMT sequential encoder and decoder are 6-layer transformer encoder and 6-layer transformer decoder respectively, which is also the same as [13]. The MT model replaced the TIMT image encoder with an embedding layer based text encoder. The sequential encoder and decoder of the MT model are kept the same as the TIMT model. The preprocessing method and experimental setting are the same as [13]. For decoding results, sacre-BLEU¹ is calculated to evaluate the translation performance.

¹ <https://github.com/mjpost/sacrebleu>.

4.3 Results of Various Knowledge Distillation

Table 1 shows the results of various knowledge distillation (KD) combinations. Line No.1, No.2, and No.4 show the results of single-teacher KD. Single decoder KD (No.1) achieves the best single-teacher performance due to the strong guidance from decoding knowledge. Sequential encoder KD (No.2) outperforms image encoder KD (No.4), indicating semantic knowledge transferring is more important for TIMT task. For bi-teacher KD comparison, sequential encoder and decoder KD combination (No.3) performs well by incorporating semantic and decoding guidance. Finally, triple-teacher KD (No.7) achieves the best performance by transferring image encoder, sequential decoder, and decoder knowledge into end-to-end TIMT model, indicating incorporating accurate knowledge into various sub-modules is vital for performance improvements.

4.4 Comparison with Existing TIMT Methods

Compared with existing end-to-end TIMT models, MTKD has significant improvements by incorporating various knowledge into sub-modules of TIMT model. Table 2 shows the comparison between MTKD and existing TIMT models. TRBA [2] is a vanilla TIR model trained with translation dataset. CLTIR [6] proposed to train TIMT model with TIR multi-task learning. RTNet [18] bridges pre-trained TIR and MT models with feature transformer. METIMT [13] trains TIMT model with MT auxiliary task. MHCMM [5] is a mimic learning based method by introducing MT teacher for TIMT model. Different from existing research, MTKD incorporates both TIR and MT teachers into TIMT optimization. Meanwhile, various knowledge distillation is utilized to transfer accurate

Table 2. Comparison of existing end-to-end models with our proposed multi-teacher knowledge distillation (MTKD) method. MTKD utilizes the knowledge distillation setting of line No.7 in Table 1.

Architecture	Synthetic			Subtitle		Street
	EnCh	EnDe	ChEn	EnCh	ChEn	ChEn
Existing End-to-End Models						
TRBA [2]	9.61	7.36	4.77	12.12	5.18	0.36
CLTIR [6]	18.02	15.55	10.74	16.47	9.04	0.43
CLTIR+TIR [6]	19.44	16.31	13.52	17.96	11.25	1.74
RTNet [18]	18.91	15.82	12.54	17.63	10.63	1.07
RTNet+TIR [18]	19.63	16.78	14.01	18.82	11.50	1.93
MTETIMT [13]	19.25	16.27	13.16	17.73	10.79	1.69
MTETIMT+MT [13]	21.96	18.84	15.62	19.17	12.11	5.84
MHCMM [5]	22.08	18.97	15.66	19.24	12.12	5.87
Our Proposed Multi-Teacher Knowledge Distillation Method						
MTKD	22.26	19.38	15.84	19.31	12.17	6.08

Table 3. Comparison of TIR+MT pipeline method with MTKD method on English-to-Chinese synthetic test set. Model size represents the parameter amount of the model. Decoding time means the time of predicting a sentence and the unit is second. BLEU score is utilized to evaluate the translation performance on valid and test set.

Architecture	Model Size↓	Decoding Time↓	Valid BLEU↑	Test BLEU↑
Pipeline	195.1M	0.33s	23.52	20.46
MTKD	121.9M	0.19s	24.13	22.26

knowledge into sub-modules of TIMT model. Finally, MTKD outperforms the existing best MHCMM model with 0.18 BLEU scores on average. Improvements in all three evaluation domains reveal the good generalization of MTKD.

4.5 Comparison with Pipeline Method

Table 3 shows the comparison of MTKD with the TIR+MT pipeline model. By transferring knowledge into TIMT model, MTKD has better translation performance, which effectively addresses the error propagation problems in pipeline model. With an end-to-end architecture, MTKD has fewer parameters than pipeline model. Meanwhile, MTKD has less decoding time than pipeline model, which is vital in real-world applications (Table 3).

4.6 Analysis of Hyper-parameter

The loss weight of knowledge distillation is a key hyper-parameter to balance the end-to-end TIMT loss and knowledge distillation losses. When $\lambda_{KD} = 0$, the

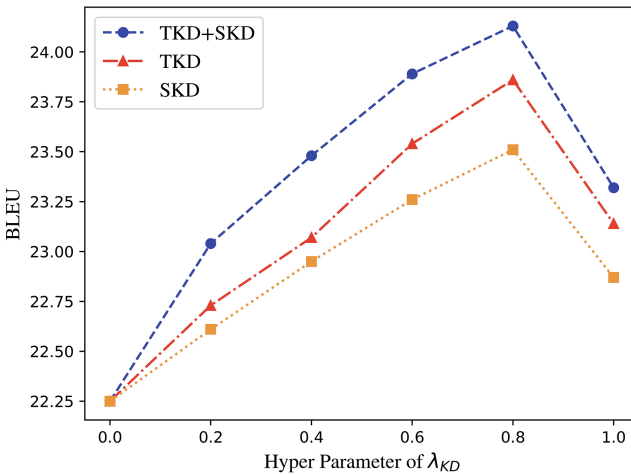


Fig. 3. Hyper-parameter analysis on the loss weight of knowledge distillation.

model is only optimized with end-to-end loss function and the performance is limited due to the end-to-end data scarcity and the difficulty of TIMT task. By incorporating KD loss, the performance is getting better and the optimal value for λ_{KD} is 0.8. When $\lambda_{KD} = 1$, the performance drops a bit, indicating end-to-end TIMT loss by guiding the model learns to predict as the ground-truth is also important for TIMT task.

5 Conclusion

In this paper, we propose a novel multi-teacher knowledge distillation (MTKD) method for end-to-end text image machine translation task. Three pre-trained teacher models are utilized to provide accurate knowledge for corresponding sub-modules in end-to-end TIMT model. By transferring various knowledge into sub-modules of TIMT model, the translation performance is significantly improved compared with existing methods. Meanwhile, token-level and sentence-level knowledge distillation are complementary for knowledge transferring, indicating that multi-granularity knowledge distillation is vital for TIMT improvements. Furthermore, MTKD based TIMT model outperforms pipeline models with a smaller model size and less decoding time, which has the advantages of both end-to-end and pipeline models. In the future, we will explore to transfer more knowledge into end-to-end TIMT model to further improve the translation performance.

Acknowledgement. This work has been supported by the National Natural Science Foundation of China (NSFC) grants 62106265.

References

1. Affli, H., Way, A.: Integrating optical character recognition and machine translation of historical documents. In: Proceedings of the Workshop on Language Technology Resources and Tools for Digital Humanities, LT4DH@COLING, pp. 109–116 (2016)
2. Baek, J., et al.: What is wrong with scene text recognition model comparisons? Dataset and model analysis. In: 2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, pp. 4714–4722 (2019)
3. Chang, Y., Chen, D., Zhang, Y., Yang, J.: An image-based automatic Arabic translation system. *Pattern Recognit.* **42**(9), 2127–2134 (2009)
4. Chen, J., Cao, H., Natarajan, P.: Integrating natural language processing with image document analysis: what we learned from two real-world applications. *Int. J. Document Anal. Recognit.* **18**(3), 235–247 (2015)
5. Chen, Z., Yin, F., Yang, Q., Liu, C.L.: Cross-lingual text image recognition via multi-hierarchy cross-modal mimic. *IEEE Trans. Multimedia (TMM)*, pp. 1–13 (2022)
6. Chen, Z., Yin, F., Zhang, X., Yang, Q., Liu, C.: Cross-lingual text image recognition via multi-task sequence to sequence learning. In: 25th International Conference on Pattern Recognition (ICPR), pp. 3122–3129 (2020)

7. Du, J., Huo, Q., Sun, L., Sun, J.: Snap and translate using windows phone. In: 2011 International Conference on Document Analysis and Recognition (ICDAR), pp. 809–813. IEEE Computer Society (2011)
8. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770–778 (2016)
9. Hinami, R., Ishiwatari, S., Yasuda, K., Matsui, Y.: Towards fully automated manga translation. In: The Thirty-Fifth AAAI Conference on Artificial Intelligence (AAAI) (2021)
10. Hinton, G.E., Vinyals, O., Dean, J.: Distilling the knowledge in a neural network. CoRR abs/1503.02531 (2015)
11. Kim, Y., Rush, A.M.: Sequence-level knowledge distillation. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, 1–4 November 2016, pp. 1317–1327. The Association for Computational Linguistics (2016)
12. Liu, Y., et al.: End-to-end speech translation with knowledge distillation. In: Interspeech 2019, 20th Annual Conference of the International Speech Communication Association, Graz, Austria, 15–19 September 2019, pp. 1128–1132. ISCA (2019)
13. Ma, C., et al.: Improving end-to-end text image translation from the auxiliary text translation task. In: 26th International Conference on Pattern Recognition, ICPR 2022, Montreal, QC, Canada, 21–25 August 2022, pp. 1664–1670. IEEE (2022)
14. Mansimov, E., Stern, M., Chen, M., Firat, O., Uszkoreit, J., Jain, P.: Towards end-to-end in-image neural machine translation. In: Proceedings of the First International Workshop on Natural Language Processing Beyond Text, pp. 70–74. Association for Computational Linguistics, Online (Nov 2020)
15. Shekar, K.C., Cross, M.A., Vasudevan, V.: Optical character recognition and neural machine translation using deep learning techniques. In: Saini, H.S., Sayal, R., Govardhan, A., Buyya, R. (eds.) Innovations in Computer Science and Engineering. LNNS, vol. 171, pp. 277–283. Springer, Singapore (2021). https://doi.org/10.1007/978-981-33-4543-0_30
16. Shi, B., Bai, X., Yao, C.: An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **39**(11), 2298–2304 (2017)
17. Shi, B., Wang, X., Lyu, P., Yao, C., Bai, X.: Robust scene text recognition with automatic rectification. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, 27–30 June 2016, pp. 4168–4176. IEEE Computer Society (2016)
18. Su, T., Liu, S., Zhou, S.: RTNet: an end-to-end method for handwritten text image translation. In: Lladós, J., Lopresti, D., Uchida, S. (eds.) ICDAR 2021. LNCS, vol. 12822, pp. 99–113. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-86331-9_7
19. Sun, H., Wang, R., Chen, K., Utiyama, M., Sumita, E., Zhao, T.: Knowledge distillation for multilingual unsupervised neural machine translation. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, 5–10 July 2020, pp. 3525–3535 (2020)
20. Sutskever, I., Vinyals, O., Le, Q.V.: Sequence to sequence learning with neural networks. In: Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, 8–13 December 2014, Montreal, Quebec, Canada, pp. 3104–3112 (2014)

21. Tan, X., Ren, Y., He, D., Qin, T., Zhao, Z., Liu, T.: Multilingual neural machine translation with knowledge distillation. In: 7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, 6–9 May 2019
22. Vaswani, A., et al.: Attention is all you need. In: Advances in Neural Information Processing Systems, pp. 5998–6008 (2017)
23. Watanabe, Y., Okada, Y., Kim, Y., Takeda, T.: Translation camera. In: Fourteenth International Conference on Pattern Recognition, ICPR 1998, Brisbane, Australia, 16–20 August 1998, pp. 613–617 (1998)
24. Weinzaepfel, P., Brégier, R., Combaluzier, H., Leroy, V., Rogez, G.: DOPE: distillation of part experts for whole-body 3D pose estimation in the wild. In: Computer Vision - ECCV 2020–16th European Conference, Glasgow, UK, 23–28 August 2020, Proceedings, Part XXVI. vol. 12371, pp. 380–397 (2020)
25. Wong, F., Chao, S., Chan, W.K.: Cyclops - snapshot translation system based on mobile device. *J. Softw.* **6**(9), 1664–1671 (2011)
26. Yang, J., Chen, X., Zhang, J., Zhang, Y., Waibel, A.: Automatic detection and translation of text from natural scenes. In: Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2002, 13–17 May 2002, Orlando, Florida, USA, pp. 2101–2104 (2002)
27. Zhang, Y., Nie, S., Liang, S., Liu, W.: Bidirectional adversarial domain adaptation with semantic consistency. In: Lin, Z. (ed.) PRCV 2019. LNCS, vol. 11859, pp. 184–198. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-31726-3_16
28. Zhang, Y., Nie, S., Liang, S., Liu, W.: Robust text image recognition via adversarial sequence-to-sequence domain adaptation. *IEEE Trans. Image Process.* **30**, 3922–3933 (2021)
29. Zhao, Y., Xiang, L., Zhu, J., Zhang, J., Zhou, Y., Zong, C.: Knowledge graph enhanced neural machine translation via multi-task learning on sub-entity granularity. In: Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, 8–13 December 2020, pp. 4495–4505 (2020)
30. Zhao, Y., Zhang, J., Zhou, Y., Zong, C.: Knowledge graphs enhanced neural machine translation. In: Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI 2020, pp. 4039–4045 (2020)