Regular Paper

Why Videos Do Not Guide Translations in Video-guided Machine Translation? An Empirical Evaluation of Video-guided Machine Translation Dataset

Zhishen Yang[1,a)] Tosho Hirasawa[2,b)] Mamoru Komachi[2,c)] Naoaki Okazaki[1,d)]

Received: September 10, 2021, Accepted: January 4, 2022

Abstract: Video-guided machine translation (VMT) is a type of multimodal machine translation that uses information from videos to guide translation. However, in the VMT 2020 challenge, adding videos only marginally improved the performance of VMT models compared to their text-only baselines. In this study, we systematically analyze why videos did not guide translation. Specifically, we evaluate the models in input degradation and visual sensitivity experiments and compare the results with a human evaluation using VATEX, which is the dataset used in the VMT 2020 challenge. The results indicate that short and straightforward video descriptions in VATEX are sufficient to perform the translations, which renders the videos redundant in the process. Based on our findings, we provide suggestions on the design of future VMT datasets. Code and human-evaluated data are publicly available for future research.

Keywords: natural language processing, multimodal machine translation, video-guided machine translation, machine translation

1. Introduction

Multimodal machine translation (MMT) extends text-only machine translation by using information from other modalities to improve translation quality. Video-guided machine translation (VMT) is a multimodal machine translation task in which videos are provided as additional inputs to the model that translates sentences from the source to target languages. Compared to image-guided machine translation, videos provide visual and acoustic modalities with rich embedded information, such as actions, objects, and temporal transitions.

VMT aims to use videos as additional information to reduce the ambiguities existing in the language, thereby improving translation quality [1]. In **Fig. 1**, VMT system could disambiguate and translate "bar" as "杆 (pullup bar)" by referring to the associated video.

The recently proposed VATEX dataset [1] is a dataset for VMT research and shared tasks. According to the results of VMT Challenge 2020, all multimodal VMT models only had marginal performance gains compared to their text-only counterparts, which contradicts the belief that videos improve translation quality. We hypothesize that this was caused by the design of the VATEX dataset: short and straightforward video descriptions are sufficient for translations, making information from the videos redundant to the models.

To examine our hypothesis, we conducted experiments with input degradation experiments, visual sensitivity, and human evaluation inspired by recent work on probing the need for visual context in image-guided machine translation [2], [3], [4].

Experiments with input degradation and visual sensitivity have two goals: 1. To examine whether VMT models can utilize videos if they provide complementary rather than redundant information. 2. To eliminate the possibility that modeling limitations and representation of videos prevent VMT models from leveraging information from videos. Human evaluation is a manual inspection to examine our hypothesis that text in the VATEX dataset is sufficient to perform the translation, and videos provide redundant rather than complementary information. The experimental and human evaluation results showed that when textual information was sufficient, visual information from videos became redundant to the VMT model.

The code used in this study and human evaluation data are publicly available [*1].

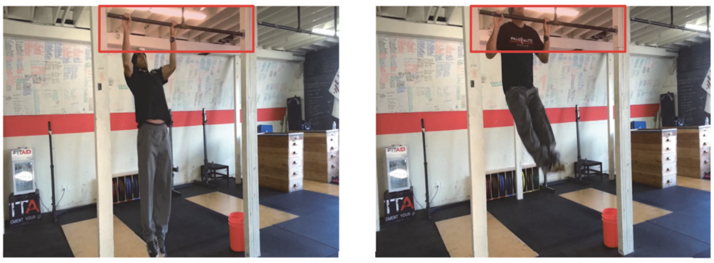
2. Related Work

2.1 Multimodal Machine Translation

Multimodal machine translation aims to generate better translation by leveraging non-linguistic information for a source sentence. The first attempt in this field focuses on using still images [5], [6], [7]. Most studies employed pretrained image clas-

[1] Tokyo Institute of Technology, Meguro, Tokyo 152–8550, Japan
[2] Tokyo Metropolitan University, Hino, Tokyo 191–0065, Japan
[a)] zhishen.yang@nlp.c.titech.ac.jp
[b)] hirasawa-tosho@ed.tmu.ac.jp
[c)] komachi@tmu.ac.jp
[d)] okazaki@c.titech.ac.jp

[*1] https://github.com/ZhishenYang/do_video_guide_translation.git



Source: A man is in a room and using **a bar** in order to do some pull ups.

Machine translation: 一个男人在一个房间里，正在使用一个**酒吧**来做一些引体向上。

Video-guided machine translation: 一个男人在一个房间里，正在使用一个**杆**来做一些引体向上。

Fig. 1 Example of VMT, where video helps to disambiguate the word "bar": "bar" should be translated as "pullup bar."

sification models, such as ResNet [8], to extract visual features from still images. The visual features are then utilized in an attentive manner [9], [10] or to initialize the decoder's hidden state [6].

Several years after the emergence of image-guided machine translation, VATEX [1] was introduced as a dataset for video-guided translation. The dataset was annotated based on a subset of Kinetics-600 [11] and comprises over 41K YouTube videos. The first VMT challenge was held in 2020. The top three teams in VMT Challenge 2020 presented recurrent neural network (RNN)-based and transformer-based models. The winning team [12] used a hierarchically attentive RNN-based model [10] with positional encoding for video features. Two other teams proposed transformer-based VMT models with modifications to incorporate video features. Although transformer-based models outperformed RNN-based models as single models, the RNN-based model achieved the best performance by combining multiple types of video features in an ensemble.

Although the hierarchically attentive RNN model and the VGT-Shallow model performed well in VMT Challenge 2020, the question raised in this study is that we are not sure whether these models use information from videos to guide translation. Therefore, we selected these two VMT models to obtain answers.

2.2 Probing Auxiliary Modalities

Probing the need for auxiliary modalities is an important topic in MMT. In image-guided machine translation, the first study [13] evaluated the usefulness of images in MMT models and concluded that MMT models did not always show high visual awareness.

In the analytic experiments in Ref. [2], source sentences were degraded in three different ways to simulate scarce textual context conditions in which images should be beneficial to translation. Recently, the benefits of multimodal inputs were explained by the results of regularization [3]. In contrast to these findings, [14] evaluated a well-established dataset of image-guided machine translation, Multi30k [15]. Human evaluation results show that 6.1% − −13.8% of English sentences require the visual context for various reasons (e.g., lexical ambiguity or an inaccurate English description), which suggests potential improvements for MMT models. In addition to MMT tasks, [16] proposed a method to isolate cross-modal interactions of multimodal classification tasks and showed that cross-modal interactions have little or no contribution to the model performance.

Extending [2], we investigate the need for videos as a visual context for translation in VMT. Our study is based on VATEX [1], which is a recently proposed dataset designed for VMT. .

3. Models

In this section, we introduce two NMT and two VMT models used in the experiments. We selected the attentive RNN (Attentive RNN) and transformer [17] as NMT models in the experiments. For the VMT models, we employed two models from VMT Challenge 2020: a hierarchically attentive RNN with positional encoding [12] and VGT-Shallow[*2].

3.1 Attentive Recurrent Neural Network

The attentive RNN is a text-only model that consists of a gated recurrent unit (GRU) [18] encoder and a conditional gated recurrent unit (CGRU) decoder [19].

Given an n-tokens input sentence $\mathbf{x} = (x_1, \cdots, x_n)$, an encoder with two stacked bidirectional GRU layers first encodes $\mathbf{x}$ into encoder states $\mathbf{h} = (h_1, \cdots, h_n)$, where each h_i is a d-dimensional vector.

The CGRU decoder consists of two unidirectional GRU layers with an attention layer in between. The CGRU receives the encoder states $\mathbf{h}$ from the GRU encoder to decode the m-tokens target sentence $\mathbf{y} = (y_1, \cdots, y_m)$. For each target position j, we use the first *GRU* layer to compute the decoder state proposal $\boldsymbol{s}_j$ from the previous word embedding w_{j-1} and the previous decoder state $\hat{\boldsymbol{s}}_{j-1}$:

$$\boldsymbol{s}_j = \mathrm{GRU}_{\mathbf{1}}(w_{j-1}, \hat{\boldsymbol{s}}_{j-1}) \tag{1}$$

Subsequently, an attention layer att that computes the textual context vector $\boldsymbol{c}_j$ along with the state proposal $\boldsymbol{s}_j$ from GRU and encoder states $\mathbf{h}$:

$$\boldsymbol{c}_j = \mathrm{att}(\boldsymbol{s}_j, \mathbf{h}) \tag{2}$$

The attention layer att computes $\boldsymbol{c}_j$ as follows:

[*2] https://www.youtube.com/watch?v=zHwXPmIQajA&t=517s

$$\boldsymbol{e}_{ij} = o^T \tanh(\boldsymbol{W} s_j + \boldsymbol{U} \boldsymbol{h}_i), \tag{3}$$

$$\alpha_{ij} = \frac{\exp(\boldsymbol{e}_{ij})}{\sum_{k=1}^{N} \exp(\boldsymbol{e}_{kj})}, \tag{4}$$

$$\boldsymbol{c}_j = \sum_{i=1}^{N} \alpha_{ij} \boldsymbol{h}_i, \tag{5}$$

where o^T, $\boldsymbol{W}$, and $\boldsymbol{U}$ are learnable projection matrices.

3.2 Transformer

The transformer [17] adopts an encoder-decoder structure. Given an n-tokens input sentence $\mathbf{x} = (x_1, \cdots, x_n)$, the encoder first encodes $\mathbf{x}$ into a representation $\mathbf{z} = (z_1, \cdots, z_n)$. The decoder then decodes $\mathbf{z}$ to generate an m-tokens target sentence $\mathbf{y} = (y_1, \cdots, y_m)$. As the transformer does not contain any recurrence or convolution, we add positional encodings to both input and output embeddings to preserve the order of sequence.

Encoder stacks N identical encoder layers, each of which contains a self-attention mechanism followed by a fully connected feed-forward network. The decoder also includes M identical decoder layers; each decoder layer has self-attention, source-target attention, and feed-forward network. Note that each sublayer in both the encoder and decoder layers is interconnected by a residual connection [8] followed by layer normalization [20].

3.3 Hierarchically Attentive Recurrent Neural Network with Positional Encoding

A hierarchically attentive RNN with positional encoding (PE) [12] is an extension of the hierarchically attentive model [10]. The underlying model adopts a simple encoder and a modified decoder from [21]. This study uses two distinct attention mechanisms to compute the textual context vector and auxiliary context vector (in our case, the context vector over sequential video representations). However, the model is assumed to incorporate spatial image features (e.g., the region of interest feature from Faster-RCNN models). Therefore, it does not leverage order information, which is a distinguishing property of video features.

To address this problem, we add positional encodings [17] to the video representations at the beginning of the attention so that the model can capture the order of the representations.

3.3.1 Encoder

First, we encode the n-tokens input sentence $\mathbf{x} = (x_1, \cdots, x_n)$ into encoder states $\mathbf{h} = (h_1, \cdots, h_n)$ using two stacked bidirectional GRU layers, where each h_i is a d-dimensional vector. The T-frame video representations $\mathbf{z} = (z_1, \cdots, z_T)$ are extracted from a video $\boldsymbol{v}$.

Moreover, we add positional encoding to the video feature $\mathbf{z}$ to obtain frame-aware video representations $\hat{\mathbf{z}} = (\hat{z}_1, \cdots, \hat{z}_T)$ at each position $pos \in (1, \cdots, T)$:

$$\hat{z}_{pos} = z_{pos} + PE_{pos}, \tag{6}$$

$$PE_{(pos,2i)} = \sin(pos/10000^{2i/d}), \tag{7}$$

$$PE_{(pos,2i+1)} = \cos(pos/10000^{2i/d}) \tag{8}$$

where i is the dimension, $pos \in (1, \cdots, T)$ is the position , and i is the dimension.

3.3.2 Decoder

For each target position j, we compute the decoder state proposal $\boldsymbol{s}_j$ from the previous word embedding w_{j-1} and the previous decoder state $\hat{\boldsymbol{s}}_{j-1}$:

$$\boldsymbol{s}_j = \mathrm{GRU}_1(w_{j-1}, \hat{\boldsymbol{s}}_{j-1}). \tag{9}$$

Subsequently, the textual context vector $\boldsymbol{c}_j^{(t)}$ and video context vector $\boldsymbol{c}_j^{(\boldsymbol{z})}$ are computed using two separate attention mechanisms, $\mathrm{att}_{\boldsymbol{t}}$ and $\mathrm{att}_{\boldsymbol{z}}$:

$$\boldsymbol{c}_j^{(\boldsymbol{t})} = \mathrm{att}_{\boldsymbol{t}}(\boldsymbol{s}_j, \mathbf{h}) \tag{10}$$

$$\boldsymbol{c}_j^{(\boldsymbol{z})} = \mathrm{att}_{\boldsymbol{z}}(\boldsymbol{s}_j, \hat{\mathbf{z}}) \tag{11}$$

The final context vector $\boldsymbol{c}_j$ is computed using another attention mechanism over modalities $m \in \{\boldsymbol{t}, \boldsymbol{z}\}$:

$$\boldsymbol{e}_j^{(m)} = o^T \tanh(\boldsymbol{W}_1 s_j + \boldsymbol{U}^{(m)} \boldsymbol{c}_j^{(m)}), \tag{12}$$

$$\alpha_j^{(m)} = \frac{\exp(\boldsymbol{e}_j^{(m)})}{\sum_{m' \in \{t,z\}} \exp(\boldsymbol{e}_j^{(m')})}, \tag{13}$$

$$\boldsymbol{c}_j = \sum_{m \in \{t,z\}} \alpha_j^{(m)} \boldsymbol{Q}^{(m)} \boldsymbol{c}_j^{(m)}, \tag{14}$$

where o^T and $\boldsymbol{W}_1$ are the model's parameters that are shared among all modalities; $\boldsymbol{U}^{(m)}$ and $\boldsymbol{Q}^{(m)}$ are dedicated model parameters for each modality; $\boldsymbol{U}^{(m)}$ and $\boldsymbol{Q}^{(m)}$ are the projection matrices that map each single-modality context vector into a common space, and o is a weight vector with the same dimensions as the common space.

The final context vector $\boldsymbol{c}_j$ is fed to the second GRU along with the decoder state proposal $\boldsymbol{s}_j$ to generate the final decoder state $\hat{\boldsymbol{s}}_j$ and output distribution $p(y_j|y_{<j})$:

$$\hat{\boldsymbol{s}}_j = \mathrm{GRU}_{\boldsymbol{2}}(\boldsymbol{c}_j, \boldsymbol{s}_j), \tag{15}$$

$$p(y_j|y_{<j}) = \mathrm{softmax}(\boldsymbol{W}_2 \hat{\boldsymbol{s}}_j + b), \tag{16}$$

where $\boldsymbol{W}_2$ and b are the model parameters.

3.4 VGT-Shallow

The VGT-Shallow first encodes an input sentence using a standard transformer encoder to retrieve the encoder state $\mathbf{h}$. Subsequently, the model employs a single fusion layer that has one visual reconstruction module, a cross-modal multihead attention module [17], and an element-wise weighted sum module. Note that we exploit normalization and residual connection between modules. Specifically, the visual reconstruction module uses multihead attention to reconstruct the auxiliary features.

$$\mathbf{h}_{\mathrm{r}}^{'} = \mathrm{multihead}_{\mathrm{r}}(\mathbf{z}, \mathbf{h}, \mathbf{h}) \tag{17}$$

where $\mathrm{multihead}_{\mathrm{r}}$ is a multihead attention module.

The obtained reconstructed feature $\mathbf{h}_{\mathrm{r}}^{'}$ is then fed into the cross-modal attention module:

$$\mathbf{h}_{\mathrm{x}}^{'} = \mathrm{multihead}_{\mathrm{x}}(\mathbf{h}, \mathbf{h}_{\mathrm{r}}^{'}, \mathbf{h}_{\mathrm{r}}^{'}) \tag{18}$$

where $\mathrm{multihead}_{\mathrm{x}}$ is a multihead attention module.

Finally, the fusion layer computes the element-wise sum over $\mathbf{h}$ using $\mathbf{h}_x^{'}$ as the weight to obtain the final multimodal representation $\mathbf{h}^{'}$:

$$\mathbf{h}^{'} = \mathbf{h}_x^{'} \odot \mathbf{h} \tag{19}$$

The model decodes the target sentence using $\mathbf{h}^{'}$ instead of $\mathbf{h}$.

3.5 Video Feature Extraction

Videos consist of a sequence of frames. We obtained two types of video frames: per-second frames and keyframes. Keyframes in videos store whole images, which often provide good visual representations of objects and scenes. In contrast, per-second frames offer more information regarding per-second visual changes in videos.

For video v, we first used ResNet-152 [8] pretrained on ImageNet [22] to extract two sets of appearance features: P_v and K_v, from per-second frames and keyframes separately.

4. Experiment

In VMT, videos function as complementary information to disambiguate translation. Therefore, VMT models should outperform text-only models. However, these two types of models have similar performances on the VATEX dataset. This section introduces two experiments that we implemented in this study: input degradation and visual sensitivity. These two experiments were conducted to determine why videos do not guide translation: whether it can be explained by the VATEX dataset, VMT models, or visual features extracted from videos.

4.1 Input Degradation

The results of VMT Challenge 2020 indicate that VMT models perform slightly better than text-only models; either VMT models cannot leverage visual information or using textual modality is sufficient to perform the translation. We conducted input degradation to examine the hypothesis that VMT models ignore the visual modality because the information provided by textual modality is sufficient to perform translation, not because of the models themselves.

Inspired by recent research on probing the need for visual context in image-guided machine translation [2], [3], [4], we conducted four source-side input degradation experiments: color, noun, verb, and progressive masking. **Table 1** shows examples of these four types of input degradation. Because these input degradation experiments simulate scarce textual information, videos should provide complementary information to VMT models. Under this condition, VMT models that rely on visual information obtained from videos should outperform text-only models that depend only on textual information.

Color Deprivation We replaced English words that represent colors in the source sentences with a special token, [c].

Noun Masking We replaced each noun in the source English sentences with a special token, [n].

Verb Masking The authors of the VATEX dataset used videos from the Kinetics-600 dataset, which contains a broad range of actions. All verbs in the source sentence were replaced with a special token, [v].

Progressive Masking Progressive-masking aims to progressively replace the last N tokens in a source sentence with a special token, [p]. Unlike other masking experiments, progressive-masking simulates a progressive low-resource scenario [2]. We hypothesize that with the increasing number of masked tokens in the source sentences, VMT models with access to visual information will perform better than text-only models. When nearly 100% of all tokens are masked, VMT models will perform video captioning with “expected length” as the only known information.

4.2 Visual Sensitivity

Videos, as another input to the VMT system, possess visual information, such as actions, objects, and scenes that can potentially guide the translation. We classify two types of video features: (1) action features derived from actions and (2) appearance features for visual objects.

In contrast to image-guided machine translation, video-guided approach uses videos as input. A video consists of a sequence of frames (still images) embedded with richer visual information than a single image. The question raised in this study is as follows: “Do visual features extracted from videos improve translation quality of VMT models?” To answer this question for video v, we also created a visual set that contained only the middle visual feature vector from P_v.

To establish a baseline in this experiment, we used randomly sampled vectors as feature vectors. For each video, we generated 10 randomly sampled vectors with the same number of dimensions as the vectors in P_v. The hypothesis is that the performance of VMT models deteriorates when the model is fed with randomly sampled vectors. **Table 2** summarizes the visual feature sets that were created for the experiments.

5. Experimental Setup

5.1 Dataset

We used VATEX v1.1 (the latest version) [*3]. Because the public test set is on hold, we split the validation set and used 50% of samples for validation and the remaining 50% for testing. The task was to translate from English to Chinese, as in VMT Challenge 2020.

Because some YouTube URLs were invalid, we downloaded 94% of the videos from the training and validation sets. **Table 3** shows the statistics of the VATEX dataset used in the experiments.

5.2 Implementation

In this section, we introduce text processing, input degradation, visual feature extraction, model configurations, and model training.

Text Processing We used spaCy [*4] to tokenize the English sentences, and then employed byte pair encoding (BPE) [23] to split the English tokens into subwords, where the number of merge operations was 8,000. Chinese translations were tokenized at the character level. The English and Chinese vocabulary sizes in the training set were 7,921 and 3,357, respectively.

[*3] https://eric-xw.github.io/vatex-website/download.html

[*4] https://spacy.io/

Table 1 Examples of input degradation.

Input Degradation	Examples
Original Text	a person drives a golf cart down the street while walking a large white dog.
Color Deprivation	a person drives a golf cart down the street while walking a large [c] dog.
Noun Masking	a [n] drives a [n] [n] down the [n] while walking a large white [n].
Verb Masking	a person [v] a golf cart down the street while [v] a large white dog.
Progressive Masking (n=6)	a person drives a golf cart down the street while [p] [p] [p] [p] [p] [p]

Table 2 Names and descriptions of video feature sets associated with each video v.

Name	Descriptions
I3D	Action feature provided by VMT Challenge 2020
Per-second frames (ResNet-152)	Appearance features extracted from per-second frames: P_v.
Keyframes (ResNet-152)	Appearance features extracted from keyframes: K_v.
Single frame (ResNet-152)	Contains only the middle visual feature vector from P_v.
Random	Contains 10 randomly sampled vectors, the same number of dimensions as vectors in P_v.

Table 3 Statistics of the VATEX dataset used in the experiments.

Split	Language	Video	Sent.	Token
Train	English Chinese	24,376	121,880	1,861,537 2,742,708
Valid	English Chinese	1,405	7,025	107,111 157,370
Test	English Chinese	1,405	7,025	106,502 157,807

Input Degradation The color deprivation masked approximately 0.4% of tokens in each of the training, validation, and test tests. For noun masking and verb masking, the corresponding numbers of tokens were approximately 28% and 14%, respectively. We selected $N = \{2, 4, 6, 8, 10, 16, 20, 30\}$ in the progressive-masking experiment. With $N = 30$, nearly 100% of the tokens were masked.

Video Feature Extraction We extracted two types of dynamic visual features from videos: action features and appearance features. The action features were extracted from videos using two-stream inflated 3D ConvNet (I3D) [24]. We used I3D features provided by VMT Challenge 2020 [*3] as action features. Each appearance feature is the averaged convolutional feature from the last convolutional layer of ResNet-152 [8] pretrained on ImageNet [22].

Model Configurations **Table 4** summarizes model configurations used in our experiments. Our implementation is based on *nmtpytorch* [25], which is a popular framework for both NMT and MMT.

Model Training We used the following loss functions: (1) negative log-likelihood loss for the RNN-based models and (2) label smoothing [26] for the transformer-based models. During the training of RNN-based models, we used the Adam optimizer with a learning rate of 0.0004, unit clipping gradient norm, 0.5 dropout rate, 0.00001 weight decay, and an early stopping patience of 10. To train the transformer-based models, we used the Adam optimizer with a learning rate of 0.0442, unit clipping gradient norm, 0.5 dropout rate, 0.00001 weight decay, and an early stopping patience of 10. For both evaluation and validation, we performed a beam search with a size of 5.

Evaluation Metric For all experiments, we used the same evaluation metric as that in VMT Challenge 2020: corpus-level BLEU [27].

6. Results

In this section, we discuss the results of the input degradation and visual sensitivity experiments, presented in **Table 5** and **Fig. 2**. Without any input degradation, all transformer-based models had higher BLEU scores than the RNN-based models. Among the VMT models, VGT-Shallow (Keyframes (ResNet-152)) achieved the best BLEU score.

In our experiments, transformer-based models had a significantly higher number of learnable parameters and, therefore, higher BLEU scores compared to the RNN-based models.

Given complete text, the performances of all VMT models did not surpass their text-only counterparts. VMT models started to benefit from using video features only when information in text was scarce. Therefore, when text contains sufficient information for translation, visual features extracted from videos were ignored by the models as noisy.

6.1 Input Degradation

In this section, we report the experimental results of four input degradation experiments: color deprivation, noun masking, verb masking, and progressive masking. Based on the experimental results, VMT models successfully leveraged information from videos and outperformed text-only models only when textual information was scarce (thus, when a larger number of words was masked).

Color Deprivation The differences between the VMT and text-only models were marginal because only a small fraction of tokens was masked, compared with models trained on complete data. Although VGT-Shallow (Keyframes (ResNet-152)) has the highest BLEU scores among all models, it is only slightly higher than its monomodal counterpart, Transformer.

Noun Masking All models had lower BLEU scores compared to those of their complete-data baselines. All VMT models, except for Hierarchically Attentive RNN (I3D) and Hierarchically Attentive RNN (random), had higher BLEU scores compared to those of the text-only models.

Noun masking has a larger masking scale; therefore, compared to the verb masking and color deprivation experiments, the VMT models can exploit the visual context to infer missing information.

Verb Masking In verb masking, all models had deteriorated

Table 4 Configurations of models used in the experiments: $D_{encoder}$, $D_{decoder}$, and $D_{embedding}$ are the dimensions of encoder, decoder, and source/target word embedding; $N_{encoder}$ and $N_{decoder}$ are the number of encoder and decoder layers; h is the number of attention heads; $N_{parameter}$ is the number of learnable parameters (using full-text and per-second frames (ResNet-152) features).

Model	$D_{encoder}$	$D_{decoder}$	$D_{embedding}$	$N_{encoder}$	$N_{decoder}$	h	$N_{parameter}$
Attentive RNN	512	512	1024	2	2		26.78M
Hierarchically Attentive RNN	512	512	1024	2	2		29.92M
Transformer	512	512	512	6	6	8	49.90M
VGT-Shallow	512	512	512	6	6	8	63.02M

Table 5 Corpus-level BLEU scores on test set. * indicates that a model is significantly different from its text-only counterpart. **bold** marks the model with the best BLEU score. *indicates statistical significance of the difference over their text-only counterparts ($p < 0.05$).

Models	T	T_{Color}	T_{Noun}	T_{Verb}
Attentive RNN	35.76	35.46	22.35	31.46
Hierarchically Attentive RNN (I3D)	35.55	35.19*	21.98*	31.50
Hierarchically Attentive RNN (Per-second frames (ResNet-152))	35.48*	35.34	24.36*	31.57
Hierarchically Attentive RNN (Keyframes (ResNet-152))	35.41*	35.21*	23.87*	31.61
Hierarchically Attentive RNN (Single frame (ResNet-152))	35.43*	35.27	23.94*	31.66
Hierarchically Attentive RNN (Random)	33.31*	35.37	22.11*	31.59
Transformer	**37.35**	36.63	23.60	32.75
VGT-Shallow (I3D)	37.13*	36.70	23.67	32.88
VGT-Shallow (Per-second frames (ResNet-152))	36.86*	36.38*	**25.15***	32.67
VGT-Shallow (Keyframes (ResNet-152))	37.23	**36.77**	23.89*	32.86
VGT-Shallow (Single frame (ResNet-152))	37.14*	36.40*	24.80*	33.04*
VGT-Shallow (Random)	36.96*	36.72	23.53	**33.17***

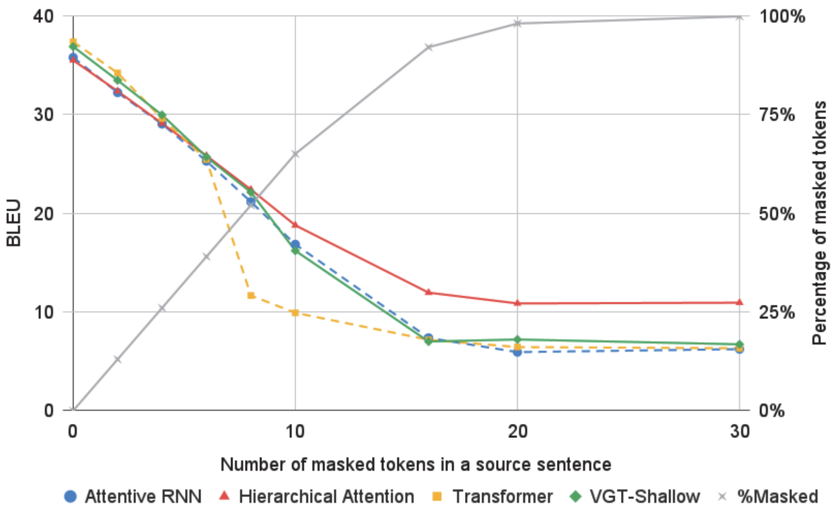


Fig. 2 Progressive Masking (using per-second frames (ResNet-152) features): the VMT models outperform the text-only models with increasing percentages of masked tokens.

performances, but the differences between the VMT and text-only models were minimal. VGT-Shallow (random) achieved the best BLEU score.

Progressive Masking Figure 2 shows the results of progressive-masking. In the case of an increasing number of masked tokens, the VMT models, especially Hierarchically Attentive RNN, started to take advantage of visual modalities and, therefore, outperformed the text-only models. Moreover, Hierarchically Attentive RNN had the best BLEU score when nearly 100% of the tokens were masked.

6.2 Visual Sensitivity

In color masking and verb masking, the VMT models had comparable performances as the models that used dynamic visual features extracted from videos using pretrained models, even when the models used randomly initialized features as visual features. We could not observe the same results in the noun masking test: when a higher number of tokens was masked, the information provided by visual features improved the model.

In the noun masking test, VGT-Shallow (Per-second frames (ResNet-152)) achieved the best BLEU score. We also found that all VMT models that used appearance features (ResNet-152) had higher BLEU scores than action features (I3D) and were statistically better than text-only models. As nouns are often related to visual objects, when a large number of nouns are masked, appearance features derived from visual objects compensate for scarce textual information, thereby helping VMT models to disambiguate translations.

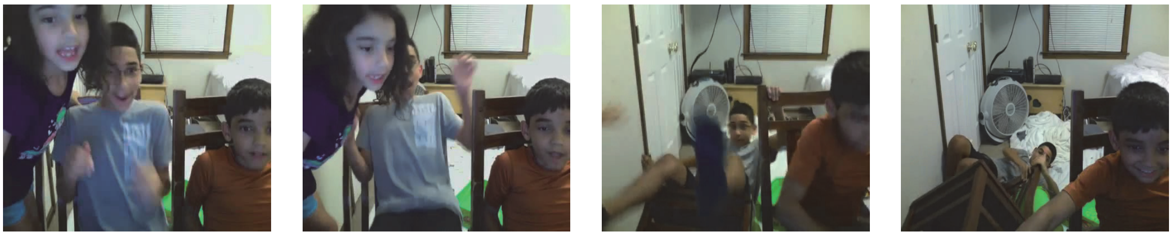
7. Human Evaluation

The authors of VATEX used a post-editing annotation strategy to collect parallel English-Chinese translation pairs, in which automatic translation systems were employed.

We hypothesize that if English-Chinese translations are good

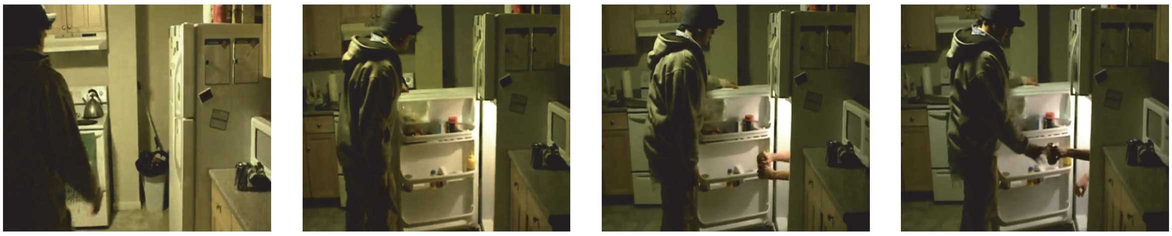
Table 6 Example of human evaluation. "Vatex translation" shows the translation from the VATEX dataset; "Text-only translation" is the human translation without showing the video; and "Post-edited translation" is the post-edited Chinese translation when showing the video. The parenthesized sentences are obtained using Google Translation of Chinese sentences.

(a) Source English description is inaccurate



English:	A person is showing three little kids making faces and being historically.
Vatex translation:	一个人正在展示三个小孩做鬼脸和历史。 (A person is showing three kids making funny faces and history.)
Text-only translation:	一个人正在展示历史上的三个做鬼脸的孩子。 (A person is showing three grimacing children in history.)
Post-edited translation:	三个孩子在电脑前，一个男孩的椅子翻了。 (Three children were in front of the computer, and one boy's chair turned over.)

(b) Wrong quantifier



English:	Young man walks into the kitchen, opens the fridge and a hand comes out of the fridge opens a can of soda and gives it to him.
Vatex translation:	年轻人走进厨房，打开冰箱，并一只手从冰箱里出来，打开一罐苏打水给年轻人。 (The young man walked into the kitchen, opened the refrigerator, and got out of the refrigerator with one hand, opening a can of soda to the young man)
Text-only translation:	年轻人走进厨房，打开冰箱，然后冰箱里出现了一只手，打开一罐苏打水并交给他。 (Young man walks into the kitchen, opens the fridge and a hand comes out of the fridge opens a can of soda and gives it to him.)
Post-edited translation:	年轻人走进厨房，打开冰箱，然后冰箱里出现了一双手，打开一罐苏打水并交给他。 (Young man walks into the kitchen, opens the fridge and a pair of hands come out of the fridge opens a can of soda and gives it to him.)

enough, videos become redundant and will not help further improve the translations in post-editing. To test our hypothesis, we conducted a human evaluation task based on a post-editing annotation strategy. The human evaluation task is to inspect data manually and examine our hypothesis that text in the VATEX dataset is sufficient to perform the translation and to check if videos provide complementary information to reduce ambiguities in translation.

7.1 Translations

We randomly selected 500 videos from the VATEX's validation set to construct the human evaluation set, and we selected two English video descriptions from each video's parallel translation pairs in random order. Therefore, the human evaluation set includes 1,000 instances; each instance has an English video description and a video URL.

Given an instance from the human evaluation set, a human translator was first asked to perform an English—Chinese translation, then watch the video, and post-edit the translation only if it does not properly describe the video. We also asked the translator to provide reasons for post-editing, as well as any remarks. Four professional translators were recruited, two for translation and two for post-editing, to guarantee translation quality.

7.2 Results

The average editing distance between original Chinese translations from VATEX and human translations is 13.3, which indicates that translations of human translators are quite different from those in VATEX. The translators post-edited 104 Chinese translations, 10.4% of the total number of instances, with an average editing distance of 5.1, compared with human translations, and 14.5 compared with original translations from the VATEX dataset. We found that 98% of post-edits were categorized as "Source English description is inaccurate," and 2% of post-edits were "Wrong quantifier," which means that the translators used information from videos to correct the wrong description in English source sentences. This result reveals the need to improve the quality of source video descriptions.

Based on the above results, we found that in most cases, source English sentences provided sufficient information for the human translator to perform translations. Videos helped to correct wrong descriptions in the source sentences rather than to disambiguate translations. These findings also indicated that the short and simple sentences in the VATEX dataset were sufficient for translation purposes, and their videos only provided redundant information, which is consistent with the automatic metrics evaluations.

7.3 Examples

Table 6 shows two examples of post-editing correction. In example (a), the edit distance between the Chinese video description in VATEX and human post-editing is 15. In the source English description, "being historically" is an ambiguous and incorrect phrase, and we cannot align it with any part of the video. Therefore, the translator changed most parts of the sentence to align with the video content. In example (b), the edit distance was 14. In the source English description, "a hand" does not match "a pair of hands" in the video; hence, the translator watched videos to correct the wrong quantifier used in the source sentence.

8. Conclusion

In this study, we investigate the dominance of textual modality and the contributions of visual modality in VMT tasks by analyzing a large-scale VMT dataset, VATEX. Results from input degradation and visual sensitivity experiments indicate that VMT models tend to ignore the visual modality when textual modality has sufficient information to perform translation. These experimental results can be explained by the fact that VATEX contains simple and short video descriptions, which provide sufficient information to accomplish translation. Hence, the visual context from videos is ignored in the translation process.

Our study intends to emphasize the need to design new datasets to further advance research on VMT. To design a new VMT dataset, we need to address the following problem of text modality dominance: when text provides sufficient information for translation, videos become redundant. Based on the empirical evaluation of VATEX and recent research on multimodal simultaneous neural machine translation, information scarcity in textual modality can allow models to exploit information from videos to improve the quality of translation. Therefore, future work on creating VMT datasets should focus on simultaneous translation of language pairs with linguistic particularities, such as different word orders and gender marking [28], [29].

Acknowledgments This work was supported by JSPS KAKENHI Grant Number 19H01118. The research results have been achieved by "Research and Development of Deep Learning Technology for Advanced Multilingual Speech Translation", the Commissioned Research of National Institute of Information and Communications Technology (NICT), Japan.

References

[1] Wang, X., Wu, J., Chen, J., Li, L., Wang, Y.-F. and Wang, W.Y.: Vatex: A large-scale, high-quality multilingual dataset for video-and-language research, *Proc. IEEE/CVF International Conference on Computer Vision*, pp.4581–4591 (2019).

[2] Caglayan, O., Madhyastha, P., Specia, L. and Barrault, L.: Probing the Need for Visual Context in Multimodal Machine Translation, *Proc. 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1* (*Long and Short Papers*), pp.4159–4170, Association for Computational Linguistics (online), DOI: 10.18653/v1/N19-1422 (2019).

[3] Wu, Z., Kong, L., Bi, W., Li, X. and Kao, B.: Good for Misconceived Reasons: An Empirical Revisiting on the Need for Visual Context in Multimodal Machine Translation, *Proc. 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing* (*Volume 1: Long Papers*), pp.6153–6166, Association for Computational Linguistics (online), DOI: 10.18653/v1/2021.acl-long.480 (2021).

[4] Li, J., Ataman, D. and Sennrich, R.: Vision Matters When It Should: Sanity Checking Multimodal Machine Translation Models, arXiv preprint arXiv:2109.03415 (2021).

[5] Hitschler, J., Schamoni, S. and Riezler, S.: Multimodal Pivots for Image Caption Translation, *Proc. 54th Annual Meeting of the Association for Computational Linguistics* (*Volume 1: Long Papers*), pp.2399–2409, Association for Computational Linguistics (online), DOI: 10.18653/v1/P16-1227 (2016).

[6] Caglayan, O., Aransa, W., Bardet, A., García-Martínez, M., Bougares, F., Barrault, L., Masana, M., Herranz, L. and van de Weijer, J.: LIUM-CVC Submissions for WMT17 Multimodal Translation Task, *Proc. 2nd Conference on Machine Translation*, pp.432–439, Association for Computational Linguistics (online), DOI: 10.18653/v1/W17-4746 (2017).

[7] Zhang, Z., Chen, K., Wang, R., Utiyama, M., Sumita, E., Li, Z. and Zhao, H.: Neural Machine Translation with Universal Visual Representation, *International Conference on Learning Representations* (2020) (online), available from ⟨https://openreview.net/forum?id=Byl8hhNYPS⟩.

[8] He, K., Zhang, X., Ren, S. and Sun, J.: Deep residual learning for image recognition, *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pp.770–778 (2016).

[9] Calixto, I., Liu, Q. and Campbell, N.: Doubly-Attentive Decoder for Multi-modal Neural Machine Translation, *Proc. 55th Annual Meeting of the Association for Computational Linguistics* (*Volume 1: Long Papers*), pp.1913–1924, Association for Computational Linguistics (online), DOI: 10.18653/v1/P17-1175 (2017).

[10] Libovický, J. and Helcl, J.: Attention Strategies for Multi-Source Sequence-to-Sequence Learning, *Proc. 55th Annual Meeting of the Association for Computational Linguistics* (*Volume 2: Short Papers*), pp.196–202, Association for Computational Linguistics (online), DOI: 10.18653/v1/P17-2031 (2017).

[11] Kay, W., Carreira, J., Simonyan, K., Zhang, B., Hillier, C., Vijayanarasimhan, S., Viola, F., Green, T., Back, T., Natsev, P., Suleyman, M. and Zisserman, A.: The Kinetics Human Action Video Dataset, *CoRR*, Vol.abs/1705.06950 (online), available from ⟨http://arxiv.org/abs/1705.06950⟩ (2017).

[12] Hirasawa, T., Yang, Z., Komachi, M. and Okazaki, N.: Keyframe Segmentation and Positional Encoding for Video-guided Machine Translation Challenge 2020, arXiv preprint arXiv:2006.12799 (2020).

[13] Elliott, D.: Adversarial Evaluation of Multimodal Machine Translation, *Proc. 2018 Conference on Empirical Methods in Natural Language Processing*, pp.2974–2978, Association for Computational Linguistics (online), DOI: 10.18653/v1/D18-1329 (2018).

[14] Frank, S., Elliott, D. and Specia, L.: Assessing multilingual multimodal image description: Studies of native speaker preferences and translator choices, *Natural Language Engineering*, Vol.24, No.3, pp.393–413 (online), DOI: 10.1017/S1351324918000074 (2018).

[15] Elliott, D., Frank, S., Sima'an, K. and Specia, L.: Multi30K: Multilingual English-German Image Descriptions, *Proc. 5th Workshop on Vision and Language*, pp.70–74, Association for Computational Linguistics (online), DOI: 10.18653/v1/W16-3210 (2016).

[16] Hessel, J. and Lee, L.: Does my multimodal model learn cross-modal interactions? It's harder to tell than you might think!, *Proc. 2020 Conference on Empirical Methods in Natural Language Processing* (*EMNLP*), pp.861–877, Association for Computational Linguistics (online), DOI: 10.18653/v1/2020.emnlp-main.62 (2020).

[17] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L. and Polosukhin, I.: Attention is All you Need, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017*, pp.5998–6008 (2017).

[18] Chung, J., Gulcehre, C., Cho, K. and Bengio, Y.: Empirical evalua-

tion of gated recurrent neural networks on sequence modeling, arXiv preprint arXiv:1412.3555 (2014).

[19] Orhanm, F. and Cho, K.: Conditional Gated Recurrent Unit with Attention Mechanism, available from ⟨https://github.com/nyu-dl/dl4mt-tutorial/blob/master/docs/cgru.pdf⟩ (2016).

[20] Ba, J.L., Kiros, J.R. and Hinton, G.E.: Layer normalization, arXiv preprint arXiv:1607.06450 (2016).

[21] Bahdanau, D., Cho, K. and Bengio, Y.: Neural Machine Translation by Jointly Learning to Align and Translate, *ICLR* (2015).

[22] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K. and Fei-Fei, L.: Imagenet: A large-scale hierarchical image database, *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp.248–255, IEEE (2009).

[23] Sennrich, R., Haddow, B. and Birch, A.: Neural Machine Translation of Rare Words with Subword Units, *Proc. 54th Annual Meeting of the Association for Computational Linguistics* (*Volume 1: Long Papers*), pp.1715–1725, Association for Computational Linguistics (online), DOI: 10.18653/v1/P16-1162 (2016).

[24] Carreira, J. and Zisserman, A.: Quo vadis, action recognition? a new model and the kinetics dataset, *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pp.6299–6308 (2017).

[25] Caglayan, O., García-Martínez, M., Bardet, A., Aransa, W., Bougares, F. and Barrault, L.: NMTPY: A Flexible Toolkit for Advanced Neural Machine Translation Systems, *Prague Bull. Math. Linguistics*, Vol.109, pp.15–28 (online), DOI: 10.1515/pralin-2017-0035 (2017).

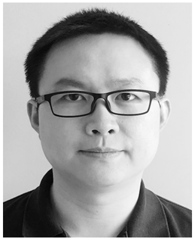
[26] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J. and Wojna, Z.: Rethinking the Inception Architecture for Computer Vision, *Proc. IEEE Conference on Computer Vision and Pattern Recognition* (*CVPR*) (2016).

[27] Papineni, K., Roukos, S., Ward, T. and Zhu, W.-J.: Bleu: A Method for Automatic Evaluation of Machine Translation, *Proc. 40th Annual Meeting of the Association for Computational Linguistics*, pp.311–318, Association for Computational Linguistics (online), DOI: 10.3115/1073083.1073135 (2002).

[28] Imankulova, A., Kaneko, M., Hirasawa, T. and Komachi, M.: Towards Multimodal Simultaneous Neural Machine Translation, *Proc. 5th Conference on Machine Translation*, pp.594–603, Association for Computational Linguistics (online), available from ⟨https://aclanthology.org/2020.wmt-1.70⟩ (2020).

[29] Caglayan, O., Ive, J., Haralampieva, V., Madhyastha, P., Barrault, L. and Specia, L.: Simultaneous Machine Translation with Visual Context, *Proc. 2020 Conference on Empirical Methods in Natural Language Processing* (*EMNLP*), pp.2350–2361, Association for Computational Linguistics (online), DOI: 10.18653/v1/2020.emnlp-main.184 (2020).

Zhishen Yang is currently a Ph.D. student at the School of Computing, Tokyo Institute of Technology, Japan. He received his master of engineering in artificial intelligence from the same institute in 2020 and his bachelor of computer science degree with first class honours (cooperative program) from Dalhousie University, Canada, in 2016. His research focuses on multimodal natural language processing.



Tosho Hirasawa received his B.S. degree from Kyoto University in 2009 and master of information science degree from Tokyo Metropolitan University in 2021. He is currently a Ph.D. Candidate at the Graduate School of System Design, Tokyo Metropolitan University. His research interests include machine translation and multimodal natural language processing.

Mamoru Komachi is an Associate Professor at Tokyo Metropolitan University (TMU). He received his M.Eng. and Ph.D. degrees from Nara Institute of Science and Technology (NAIST) in 2007 and 2010. He was an Assistant Professor at NAIST before joining TMU. His research interests include semantics, information extraction and educational applications of natural language processing.

Naoaki Okazaki is a professor in School of Computing, Tokyo Institute of Technology, Japan. Prior to this faculty position, he worked as a post-doctoral researcher in University of Tokyo (2007–2011), and as an associate professor in Tohoku University (2011–2017). He is also a visiting research scholar of the Artificial Intelligence Research Center (AIRC), National Institute of Advanced Industrial Science and Technology (AIST). His research areas include Natural Language Processing (NLP), Artificial Intelligence (AI), and Machine Learning.

(Editor in Charge: *Kazuhiro Seki*)