

# The Role of the Input in Natural Language Video Description

Silvia Cascianelli, *Student Member, IEEE*, Gabriele Costante, *Member, IEEE*, Alessandro Devo, Thomas A. Ciarfuglia, *Member, IEEE*, Paolo Valigi, *Member, IEEE*, and Mario L. Fravolini

**Abstract**—Natural Language Video Description (NLVD) has recently received strong interest in the Computer Vision, Natural Language Processing (NLP), Multimedia, and Autonomous Robotics communities. The State-of-the-Art (SotA) approaches obtained remarkable results when tested on the benchmark datasets. However, those approaches poorly generalize to new datasets. In addition, none of the existing works focus on the processing of the input to the NLVD systems, which is both visual and textual. In this work, it is presented an extensive study dealing with the role of the visual input, evaluated with respect to the overall NLP performance. This is achieved performing data augmentation of the visual component, applying common transformations to model camera distortions, noise, lighting, and camera positioning, that are typical in real-world operative scenarios. A t-SNE based analysis is proposed to evaluate the effects of the considered transformations on the overall visual data distribution. For this study, it is considered the English subset of Microsoft Research Video Description (MSVD) dataset, which is used commonly for NLVD. It was observed that this dataset contains a relevant amount of syntactic and semantic errors. These errors have been amended manually, and the new version of the dataset (called MSVD-v2) is used in the experimentation. The MSVD-v2 dataset is released to help to gain insight into the NLVD problem.

**Index Terms**—Video Description, Multimodal Data, Input Preprocessing.

## I. INTRODUCTION

VISUAL AND TEXTUAL data-based tasks [1] are receiving growing interest in many research communities. Some studied problems are visual content retrieval based on natural language queries [2]–[5], text-guided video summarization [6], [7], story understanding [8], and visual content description [9]–[11]. This paper tackles the video description problem (NLVD). This is particularly interesting both for its research challenges and for its numerous possible applications. These include automatic video captioning of web content, automatic generation of the Descriptive Video Service (DVS) track of movies, products for the visually impaired and the blind, effective human-machine interaction, service and collaborative robotics applications, and video surveillance to name a few. The approaches developed to address this problem are data-driven. In the training phase, the NLVD systems receive as input a video stream and an associated description, that is a sentence in natural language. In the test phase, those

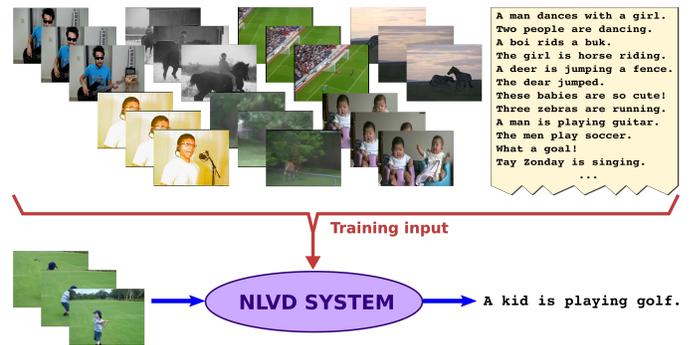


Fig. 1: Natural Language Video Description systems are trained on videos and associated captions. In the test phase, these systems are expected to produce a relevant and syntactically correct sentence describing unseen videos.

systems are expected to output a descriptive sentence given a video (Fig. 1). The quality of the produced description is difficult to assess objectively [12], [13]. Nevertheless, to obtain a quantitative evaluation, the common practice is adopting metrics designed for NLP tasks such as machine translation and summarization, and for image description. The SotA approaches obtained good results on the benchmark datasets in terms of evaluation metrics. However, the human performance in terms of the same metrics is still significantly higher (see TABLE IV). Another issue with the current NLVD methods is that both training and test are performed on the same dataset. The recent work by Cascianelli *et al.* [14] outlined the poor generalization capabilities of those algorithms when tested on a new dataset. This may limit their practical applicability.

The recently growing interest in NLVD is accompanied by intense activity of design and collection of new datasets suitable for studying the problem. The most commonly used datasets for NLVD are the Montreal Video Annotation dataset (M-VAD) [15], the Max Plank Institute of Informatics Movie Description dataset (MPII-MD) [16], the Microsoft Video Description Corpus (MSVD) [17] and the Microsoft Research - Video to Text dataset (MSR-VTT) [18]. These datasets are generic in the depicted actions and featured actors in the scene. The M-VAD and the MPII-MD contain snippets from movies, which typically have high resolution. The MSVD and the MSR-VTT, instead, include videos from YouTube, which thus have a more varied quality. In this respect, these two latter datasets seem more suitable for the study of NLVD systems able to generalize. However, it is not guaranteed that they capture the high variability of the video quality

The authors are with the Department of Engineering, University of Perugia, Perugia, Italy (e-mail: silvia.cascianelli@unipg.it; gabriele.costante@unipg.it; alessandro.devo@studenti.unipg.it; thomas.ciarfuglia@unipg.it; paolo.valigi@unipg.it; mario.fravolini@unipg.it)

Manuscript received Month XX, 2018; revised Month XX, 201X.

(e.g., color channels, resolution) in the problem. This is an obstacle to the deploy of NLVD systems in applications such as surveillance and service robotics, where the characteristics of the camera and its position in the scene differ from scenario to scenario. From a textual standpoint, in the M-VAD and MPII-MD, the videos are paired with the associated sentence from the script or the transcribed DVS track. Therefore, these datasets lack in diversity of the possible description for each video. This also has a drawback in the evaluation procedure since using the standard evaluation metrics can be to some extent misleading [19]. In the MSVD and MSR-VTT, there are several descriptions for each video (on average 43 in the first dataset, 20 in the other,) collected via the Amazon Mechanical Turk (AMT) service. Since they better capture the different ways to describe the same video, these two datasets seem more suitable to study NLVD. However, not only SotA methods still perform poorly on them, but also humans obtain not perfect performance scores (see TABLE IV and TABLE VII). In the sight of these considerations, this study is conducted using the MSVD dataset.

It is well known that the quality of the training data is crucial for the performance of NLVD algorithms. Therefore, it is important to use the most reliable datasets, deeply analyse their characteristics, and design the training input properly. Input preprocessing is a well-known good practice for effectively training machine learning algorithms [20]–[22]. For example, via data augmentation the training set can be automatically enlarged, thus providing more samples to the algorithm. This reduces the overfitting and increases the generalization capability of the model. Further, via data cleansing outliers and incorrect samples are removed, thus the distribution of the dataset should better represent the problem. This reduces the training time and increases the accuracy of the models. To the best of our knowledge, the role of the input has been neglected so far for NLVD systems. In our opinion, this aspect should be deeply explored for two main reasons: to allow improving the generalization capabilities and to gain further insights into the problem and thus design NLVD algorithms more judiciously. In the sight of this, the purpose of this work is to tackle the following practical issues: 1) to quantify the performance improvement due to input preprocessing; 2) to provide some practical guidelines for a rational selection of suitable input augmentation strategies.

For this study, the benchmark MSVD dataset is considered, and a standard encoder-decoder NLVD system is designed. A number of visual transformations are then applied to the videos in the dataset. The selection of the most appropriate appearance transformations for visual data augmentation is guided both by a data-driven analysis based on t-SNE [23]. Further, since the transformed videos have to preserve the original semantic content, the augmentation strategies have been selected among those that do not affect the relation between the video and the associated description. In the experimentation, it was observed that the MSVD dataset contains a relevant number of syntactic and semantic errors. This suggested to (manually) amend these inconsistencies producing an improved dataset, called MSVD-v2. This new dataset is used in addition to the original one in the experiments, to evaluate the effects of

training the NLVD system with more consistent textual data.

The remainder of the paper is organized as follows. In Section II the related work is overviewed. In Section III the proposed approach is explained. In Section IV the result of an extensive experimental study are reported and discussed. In Section V the conclusions are traced.

## II. RELATED WORK

The NLVD problem is attracting the interest of many research communities, from the Computer Vision [15] and NLP [23] community to the Multimedia [10], [11], [24] and Autonomous Robotics ones [14]. The early proposed approaches to NLVD consist in addressing the task as template filling [17], [23], [25] or description retrieval [11], [26].

The most recent and most popular approach to NLVD is treating the problem as a machine translation one [27], from a video sequence to a natural language sentence, using and encoder-decoder architecture. The frames of the video are usually subsampled and processed by one or more Convolutional Neural Network (ConvNet) to extract a visual descriptor for the frame. Object recognition ConvNets and action recognition ConvNets are commonly used and combined together to obtain a good representation of the frames. Integrating the Optical Flow is also a used strategy [28]. Another recently proposed approach [29] consists in representing the video frames via a sequential vector of locally aggregated descriptor (SeqVLAD) layer, that combines a VLAD encoding and a recurrent-convolutional network. The SeqVLAD framework aggregates the intra-frame spatial information and the inter-frame motion information. The frames descriptors are used to encode the video. The encoding can be obtained directly by mean pooling the features, as done, e.g., in [30], or, more effectively, via an RNN-based encoder. Typically, is used an LSTM-based encoder. This can be a single LSTM [31], a bidirectional LSTM (BiLSTM) [32], or a multilayer LSTM [33]. Using the GRU in the encoder is less common [14]. The video encoding is then fed to the sentence decoder together with the ground truth sentence, word-by-word. The words in the ground truth sentences are used to form a vocabulary for the dataset. The words in the caption are represented as vectors in a Word Embedding (WE) [34], [35]. The WE is usually learned during the training of the NLVD system [36], or in some cases is a pretrained WE, as in [37]. The decoder is trained to predict the probability of each word in the vocabulary to be the next one in the sentence based on the video encoding and the previous words in the sentence. At each step, the most probable word is emitted, and the process stops when an End-Of-Sequence (<EOS>) tag is emitted. The decoder is designed to be a recurrent architecture. The LSTM is the preferred choice, either as a single block [19] or in a multilayer LSTM-based architecture [28]. Some works [14], [33], [38] employ the GRU as the main block of the decoder.

To improve the performance, attention mechanisms are employed at different points of the encoder-decoder system. In particular, at each word generation step, the decoder takes as input the video features weighted according to their relevance to the next word, based on the previously emitted words

[31], [32], [38], [39]. With the same principle, in [40] the attention mechanism is applied to the mean-pooled features from a predefined number of objects tracklets in the video. In [41], the textual information is used to select Regions-of-Interest (ROIs) in the video frames, whose descriptors are combined with those of the global frame in a Dual Memory Recurrent Model. An alternative strategy to combine visual and textual information is reshaping the feature vectors into circulant matrices and combining them to extract the multimodal relation among the two different modalities [42], or building multimodal matching tensor of sequential data [43]. The attention mechanism can be implemented as an additional layer in the encoder-decoder architecture or can be integrated into the gating strategy of the decoder, as done in [10]. Recent trends include training multitask NLVD models [44], [45], using a reinforcement-learning framework [46], [47], or a cycle learning framework [48].

Devising a SotA NLVD system is beyond the scope of this paper. Here, the focus is on the input to these systems and the effects of its preprocessing on the NLVD performance. The study is conducted considering a simple yet effective NLVD encoder-decoder architecture.

*a) Input Preprocessing:* Data-driven approaches, such as Deep Learning-based ones, heavily depend on the quality of the training data, in terms of effectiveness, achieved representation power, and generalization capability. For this reason, attention is usually put on properly preprocessing the input to those algorithms [21]. Data augmentation at the visual level is a well-known strategy to improve the performance of algorithms for many Computer Vision tasks. Emblematic is the case of [49], where the generalization capabilities of the AlexNet ConvNet increased by training the model on altered images. To be beneficial for the training, the applied alterations should be carefully designed to capture the characteristics of the data of the problem. In this work, it is proposed for the first time visual data augmentation for NLVD, taking into account the characteristics of the videos captured by the camera in various application scenarios, and maintaining the relation with the associated descriptions.

In the recent work in [50], it is presented style augmentation as a novel strategy to perform visual data augmentation exploiting a style transfer network [51]. In particular, the texture, contrast, colour and illumination of the image is altered, but shapes and semantic content are preserved. This strategy has been found effective for improving the performance on classification tasks, domain transfer and depth estimation. Style transfer via neural networks was introduced by Gatys *et al.* in [52], and many other works followed this approach for transforming images with the style of paintings [51], [53], [54] or other photorealistic images taken under completely different conditions [55]. The content representation and the style representation of the input image are extracted from a pre-trained ConvNet. In particular, the content is represented by the feature responses in higher layers, and the style is represented by the feature correlations of multiple lower layers. Content and style are modelled by two separate terms of the loss function, minimized to synthesise the new image having the desired style and content. Following the novel approach

of [50], in this work style augmentation is tested for NLVD.

In the NLP literature, data augmentation has been proposed to enlarge the training corpora automatically. For example, the authors of [56] performed textual data augmentation by replacing words with their synonyms from WordNet [57] for ConvNet-based models for ontology classification, sentiment analysis, and text categorization. In [58] the focus was on Natural Language Normalization and it was addressed the problem of small datasets for that. The authors trained a machine translation architecture on a small normalization dataset and translated in an unnormalized form a bigger corpus of standard text. With this, the authors were able to augment the small text normalization datasets. In [59], data augmentation for machine translation was performed, targeting rare words. The authors trained an LSTM language model to alter both source and target sentences in a parallel corpus. This way, they maintained the relation between the two sentences in the two languages. Doing the same for NLVD is not straightforward because one of the two "languages" is visual. Few works on NLVD operate at input level. In [40] data augmentation is proposed at the sentence level. The authors proposed to enrich the sentence part of the MSR-VTT with sentences from the MSVD. These sentences are selected based on the visual similarity between the associated videos in the two datasets. However, once included in the MSR-VTT, the sentences are paired with fake videos, *i.e.*, all-zeros vectors. Thus, this approach does not maintain the relation between video and text. In this paper, a new version of a benchmark dataset is presented. The sentences associated with the videos have been manually checked and corrected in case of errors, thus maintaining their semantic relatedness to the videos.

### III. PROPOSED APPROACH

To study the role of the input in the NLVD problem a basic encoder-decoder architecture is designed, and a standard benchmark dataset, namely the MSVD [17], isn considered. In this section, it is described the NLVD system, the video augmentation strategy, and the text checking procedure that led to the amended version of the dataset.

First of all, it is instructive to briefly overview the standard evaluation metrics used for NLVD systems and throughout this study to guide the design choices. These metrics are: BLEU [60], in its 4-gram variant; ROUGE [61] in its Longest Common Subsequence (LCS) variant; METEOR [62]; CIDEr [12].

Call  $n$ -gram a sequence of  $n$  consecutive words. Given a candidate sentence A and a reference sentence B to compare:

- The ratio of the number of  $n$ -grams in A that are mapped to  $n$ -grams in B to the total number of  $n$ -grams in A is the  $n$ -gram precision.
- The ratio of the number of  $n$ -grams in A that are mapped to  $n$ -grams in B to the total number of  $n$ -grams in B is the  $n$ -gram recall.

BLEU is a precision-oriented metric designed for machine translation evaluation. To obtain the score,  $n$ -gram precision is calculated considering  $n$ -grams up to length four. BLEU correlates well with human judgement on the quality of the

translation when evaluated on the entire test set, but poorly at the sentence level.

ROUGE is a recall-oriented metric designed for summarization evaluation. It is based on the idea that a candidate summary should ideally overlap the reference summary. This metric has three variants, depending on the sentences comparison strategy adopted. In the NLVD literature, it is used the variant that considers the longest common sequence (LCS), called ROUGE<sub>L</sub>. All ROUGE variants correlate well with human judgement.

METEOR is a precision and recall-based MT evaluation metric. For its computation, unigrams in the candidate and reference sentences are matched based on their exact form, *i.e.*, if the unigrams are the same word, stemmed form, *i.e.*, if the unigrams have the same root, and meaning, *i.e.*, if the unigrams are synonyms. Then, unigram precision and unigram recall are calculated based on the found matches, and the F-mean is obtained, weighing the recall more than the precision. In addition, a multiplicative factor is used to reward identically ordered contiguous matched unigrams. METEOR correlates better than unigram precision, unigram recall and their harmonic combination, with human judgement also at the sentence level.

CIDEr is a metric designed to assess the quality of the description of an image. It is based on the cosine similarity between  $n$ -grams in the candidate description and in the set of reference descriptions associated to the image. Each  $n$ -gram is weighted using a Term Frequency-Inverse Document Frequency (TF-IDF) strategy. This metric is designed to correlate well with human judgement on the image description quality, thus is particularly suitable for the task of NLVD.

The possible values for all the above metrics span from 0 to 1. For all but CIDEr, these are reported using values from 0 to 100. The values of the CIDEr metric are reported between 0 and 1000. This is done to make the CIDEr values of the same order of magnitude as those of the other metrics. In fact, even SotA approaches obtain very low scores in terms of the CIDEr metric.

#### A. Basic Encoder-Decoder NLVD System

Outperforming the SotA is beyond the scope of this paper, thus a simple yet effective encoder-decoder architecture is designed and used. This helps in better highlighting the effects of the input preprocessing on the performance. Its pictorial representation is in Fig. 2. In the following, the model is referred to as Basic Encoder-Decoder Description System (BEDDS).

The video frames are sampled one every five. On the sampled frames, the output of the last fully connected layer of the *ResNet50* [63] and the *C3D* [64] ConvNets is computed. The choice of these two SotA ConvNets is the result of a preliminary ablation study and confirms the results reported, *e.g.*, in [28], [48] on the benefits of using very deep object recognition ConvNets and including the temporal information either via action recognition ConvNets or Optical Flow. This allows capturing both the appearance and the movement in the frame. In this study, it has been used *ResNet50* instead

	$B_4$	$R_L$	$M$	$C$
BEDDS (VGG16) + WE + VE	41.5	66.8	30.4	60.7
BEDDS (VGG16) + WE	41.2	67.0	30.9	57.1
BEDDS (VGG16) + WE - GRU enc.	41.9	67.5	30.6	54.1
BEDDS (ResNet50+C3D) + WE	45.0	69.2	32.3	66.7
BEDDS (ResNet50+C3D) + WE - GRU enc.	43.9	69.1	<b>32.9</b>	69.9
BEDDS (ResNet50+C3D) + GloVe finetuned	43.6	69.0	32.3	69.9
BEDDS (used for the study)	<b>45.1</b>	<b>69.4</b>	<b>32.9</b>	<b>70.0</b>

TABLE I: Preliminary ablation study on the encoder-decoder architecture used for this study on the MSVD.  $B_4$  stands for BLEU<sub>4</sub>,  $R_L$  for ROUGE<sub>L</sub>,  $M$  for METEOR, and  $C$  for CIDEr. Bold indicates the best performance.

of its deeper variants to limit the computational cost of the experiments. Note that for the *C3D* vector it is considered a sliding window centered in the sampled frame containing 16 frames. The outputs of the ConvNets are concatenated to form the feature vector  $\mathbf{x}_*$  describing the frame. This vector is 2048+4096-dimensional. As a result, the input video is represented by the sequence of feature vectors describing its  $N$  frames ( $\mathbf{x}_0, \mathbf{x}_1, \dots, \mathbf{x}_N$ ). Usually, in the NLVD literature, the visual feature vectors are mapped in a lower dimensional space via a learnt linear transformation (VE). In the NLVD architecture used for this study, it has been decided not to perform this mapping operation in the sight of the preliminary study whose results are reported in TABLE I.

The sentence words are converted to lower-case, and the punctuation is removed. The Begin-Of-Sequence (<BOS>) and the <EOS> tags are prepended and appended respectively to the sentence. Afterwards, the so preprocessed sentence is tokenized, and the tokens form the dataset vocabulary  $D$ . Some SotA NLVD approaches include in the vocabulary only those words that appear in the dataset with a minimum frequency. For this study, it is decided to include all the words in the vocabulary to exclude the effects of the additional minimum frequency hyperparameter on the performance. The words in the dataset are represented using the 300-dimensional GloVe [34], [35] WE, pre-trained on a six billion words corpus. In many SotA architectures the WE is learned from scratch or a pretrained WE is finetuned during the training of the NLVD system. In this study, all these strategies for the WE have been tested, and the pretrained GloVe WE led to the best performance (see TABLE I.) In addition, with this choice, the overall model has fewer parameters to train. Note that in case a word in the dataset has not a corresponding vector in the GloVe embedding, a 300-dimensional random valued vector is assigned to it. In general, such words are either proper nouns, typos or very rare words. In fact, their amount decreases from  $\sim 2600$  to  $\sim 130$  after the textual data cleansing procedure described in III-C. As a result, the input sentence is represented by the sequence of embedding vectors corresponding to its  $L$  words ( $\mathbf{y}_0, \mathbf{y}_1, \dots, \mathbf{y}_L$ ).

The frames feature vectors are fed, one at a time, to the encoder LSTM [65]. Using the LSTM as the main block of the encoder in the NLVD systems is a common practice. In the case of this study, the choice was guided by a preliminary study in which the LSTM and the GRU have been compared as main block of the encoder. The study (see TABLE I) confirmed the results of [14] in that the two blocks are equivalent in terms

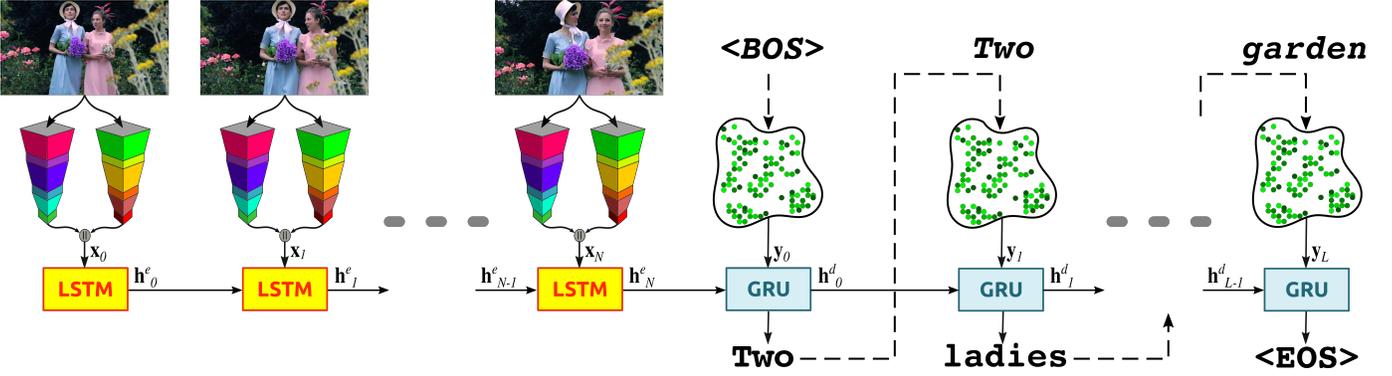


Fig. 2: Architecture of the encoder-decoder model used in this study. Recurrent layers are depicted as unfolded graphs for explanatory purpose. The *ResNet50* and *C3D* ConvNets extracts features from the video frames, which are the input to the LSTM encoder. The final state of the encoder and the GloVe embedding of the words in the caption are the input to the GRU decoder, which generates the output description one word at a time until it emits the  $\langle \text{EOS} \rangle$  tag.

of overall performance. Although GRU has fewer parameters, for this work it is decided to use an LSTM-based encoder, because this is the common strategy in the NLVD literature. The LSTM is a Deep-RNN able to handle both long and short-term temporal dependencies between serial data. It has an inner memory cell  $\mathbf{c}_n$  and a gating strategy to update the memory cell value and produce the output  $\mathbf{h}_n^e$ , based on the current input  $\mathbf{x}_n$ , and the previous state  $\mathbf{c}_{n-1}$  and output  $\mathbf{h}_{n-1}^e$ . In particular, the new memory cell value is obtained by combining the previous value, multiplied by the forget gate  $\mathbf{f}_n$ , and a candidate new state  $\tilde{\mathbf{c}}_n$ , multiplied by the input gate  $\mathbf{i}_n$ . This is to modulate how much to forget the previous value, and how much to update the current value with the new information. The output is obtained by multiplying the current memory cell with the output gate, that modulates how much memory to expose for the output. More formally, the LSTM used as the encoder in this study is defined by the following equations (1)-(6).

$$\mathbf{f}_n = \sigma(W_f \mathbf{x}_n + U_f \mathbf{h}_{n-1}^e + \mathbf{b}_f) \quad (1)$$

$$\mathbf{i}_n = \sigma(W_i \mathbf{x}_n + U_i \mathbf{h}_{n-1}^e + \mathbf{b}_i) \quad (2)$$

$$\mathbf{o}_n = \sigma(W_o \mathbf{x}_n + U_o \mathbf{h}_{n-1}^e + \mathbf{b}_o) \quad (3)$$

$$\tilde{\mathbf{c}}_n = \tanh(W_c \mathbf{x}_n + U_c \mathbf{h}_{n-1}^e + \mathbf{b}_c) \quad (4)$$

$$\mathbf{c}_n = \mathbf{f}_n \odot \mathbf{c}_{n-1} + \mathbf{i}_n \odot \tilde{\mathbf{c}}_n \quad (5)$$

$$\mathbf{h}_n^e = \mathbf{o}_n \odot \tanh(\mathbf{c}_n) \quad (6)$$

where the  $W_*$ s, the  $U_*$ s, and  $\mathbf{b}_*$ s are learnable weight matrices and bias vectors,  $\sigma$  is the sigmoid function,  $\tanh$  is the hyperbolic tangent function, and  $\odot$  is the element-wise product.

The last output of the encoder, which represents the entire video, is passed to the decoder as its first state, *i.e.*,  $\mathbf{h}_N^e = \mathbf{h}_0^d \doteq \mathbf{v}$ . The first input to the decoder is the WE of the  $\langle \text{BOS} \rangle$  token, the subsequent inputs are the WE of the words in the sentence, which terminates with the WE of the  $\langle \text{EOS} \rangle$  token. In this work, the decoder is a GRU [66]. The GRU is a more recent Deep-RNN, able to deal with both long and short-term time dependencies between the elements in a sequence. Different from the LSTM, it has not a memory cell, and its output corresponds to its inner state  $\mathbf{h}_l^d$ . Similar to the

LSTM, the state is calculated via a gating strategy using the current input  $\mathbf{y}_l$  and the previous state  $\mathbf{h}_{l-1}^d$ . First, a candidate state  $\tilde{\mathbf{h}}_l^d$  is computed from the current input and the previous state, multiplied by the reset gate  $\mathbf{r}_l$ . This gate controls how much of the old state to forget in the candidate new state. Afterwards, the state is updated, also obtaining the output. To this end, the previous state and current candidate state are combined after being multiplied by the update gate  $\mathbf{z}_l$ . This gate controls how much of the old state to maintain and how much of the current candidate state to use in the new state. More formally, the GRU used as the decoder in this study is defined by the following equations (7)-(10).

$$\mathbf{r}_l = \sigma(W_r \mathbf{y}_l + U_r \mathbf{h}_{l-1}^d + \mathbf{b}_r) \quad (7)$$

$$\mathbf{z}_l = \sigma(W_z \mathbf{y}_l + U_z \mathbf{h}_{l-1}^d + \mathbf{b}_z) \quad (8)$$

$$\tilde{\mathbf{h}}_l^d = \tanh(W_h \mathbf{y}_l + U_h (\mathbf{r}_l \odot \mathbf{h}_{l-1}^d) + \mathbf{b}_h) \quad (9)$$

$$\mathbf{h}_l^d = (1 - \mathbf{z}_l) \odot \mathbf{h}_{l-1}^d + \mathbf{z}_l \odot \tilde{\mathbf{h}}_l^d \quad (10)$$

where the  $W_*$ s, the  $U_*$ s, and  $\mathbf{b}_*$ s are learnable weight matrices and bias vectors.

At each step, the decoder outputs the state  $\mathbf{h}_l^d$ . This is multiplied by a weight matrix  $W_D$  to obtain the output vector  $\hat{\mathbf{y}}_l$ . From this, the output word is selected from the vocabulary using the softmax function, that models the probability that the output word is the next one in the description, *i.e.*, :

$$Pr(\hat{\mathbf{y}}_l | \mathbf{v}, \mathbf{y}_0, \mathbf{y}_1, \dots, \mathbf{y}_{l-1}) \sim \frac{e^{\hat{\mathbf{y}}_l}}{\sum_{\mathbf{y} \in D} e^{\mathbf{y}}} \quad (11)$$

In the training phase, the objective is to maximize the log-likelihood of the words over the sentence, *i.e.*,

$$\max_{\mathbf{W}} \sum_{l=1}^L \log Pr(\hat{\mathbf{y}}_l | \mathbf{v}, \mathbf{y}_0, \mathbf{y}_1, \dots, \mathbf{y}_{l-1}) \quad (12)$$

where  $\mathbf{W}$  denotes all the parameters of the model.

In the test phase, the input to the GRU decoder at each step is the previous word emitted, and the decoding process stops automatically when the  $\langle \text{EOS} \rangle$  token is emitted as the most probable token.

Training videos	Training samples	$B_4$	$R_L$	$M$	$C$
200	8182	33.0	63.7	27.6	36.3
400	16221	35.4	65.6	29.3	48.7
600	24599	41.6	66.3	29.6	52.4
800	32606	42.0	68.0	31.1	58.4
1000	41035	44.8	69.2	32.5	68.1
1200	49158	<b>45.1</b>	<b>69.4</b>	<b>32.9</b>	<b>70.0</b>

TABLE II: BEDDS model performance on the MSVD depending on the number of training videos.  $B_4$  stands for BLEU<sub>4</sub>,  $R_L$  for ROUGE<sub>L</sub>,  $M$  for METEOR, and  $C$  for CIDEr. Bold indicates the best performance.

### B. Data Augmentation to Study the Role of the Visual Input

For the training of Deep Neural Networks, the availability of a large number of training samples is critical. NLVD systems are no exception, as it can be read from TABLE II. This is a first reason to perform data augmentation for enlarging the training set via videos alteration. In addition, these systems lack in the generalization capability, as observed from [14].

In this work, to study the generalization capabilities and the performance of the designed NLVD system under not ideal characteristics of the visual input, it is proposed to apply alterations to the videos in the MSVD that could reflect some operating conditions of cameras in real-world scenarios. In fact, when the video captioning system is used in a specific application context (*e.g.*, for a greyscale surveillance camera placed above the monitored scene) some considerations on the characteristics of the images can be traced (the images will be greyscaled, keystone distorted, possibly blurred, occasionally very dark or very bright, etc.) According to those considerations, altered videos can be included in the dataset used for training or fine-tuning the captioning system.

The transformations applied in this study are the following:

- Grayscale conversion, to model grayscale cameras, which are largely used *e.g.*, for surveillance and robotics applications.
- Gaussian blur, to model the occasional out-of-focus operating condition.
- Keystone distortion, to model the not optimal position of the camera in the scene. In fact, *e.g.*, flying drones and surveillance cameras are usually above the scene, while *e.g.*, small terrestrial robots, kids, or users seated underneath a stage are below the scene. In this work, this distortion has been implemented using the perspective transform.
- Brightness enhancement and reduction, to model the different illumination conditions that may be encountered in the application scenario.
- Salt & Pepper noise, to generally represent low-quality images from cheap cameras.

Each of them is applied, to different degrees of severity, to all the videos on the MSVD. Exemplar applied transformations are reported in Fig. 3.

In addition, the videos have been transformed in the style of some artistic paintings. This was motivated by the fact that the strategy to perform data augmentation via style transfer has been found beneficial for many Computer Vision tasks

[50]. In this study, the effectiveness of style augmentation for NLVD is investigated. In particular, the approach of [67] has been adopted to transform the videos directly. The applied approach builds on the style transformation network in [53] and uses the instance normalization proposed in [54] instead of batch normalization. The artistic styles selected are those of Pablo Picasso’s ‘La Muse’, Leonid Afremov’s ‘Rain Princess’, Edvard Munch’s ‘The Scream’, Francis Picabia’s ‘Unnie’, Katsushika Hokusai’s ‘The Great Wave off Kanagawa’, and William Turner’s ‘The Wreck of a Transport Ship’. Some examples are reported in Fig. 4.

The applied alterations, either classical or artistic, do not modify the semantic content of the video. In fact, when selecting the transformations, those that would have affected the semantic content of the video have not been considered. For example, cropping, which is a typically applied visual data augmentation strategy, has not been considered to avoid the risk of cropping out something described in the caption.

### C. Data Cleansing to Study the Role of the Textual Input

Providing high-quality training sentences to NLVD models is critical to achieving good performance. Among the available datasets for studying NLVD, in this study the popular MSVD [17] is adopted. This is the English portion of the dataset presented by Chen and Dolan [68] for paraphrase evaluation. The authors asked AMT workers to describe with a complete, grammatically-correct sentence a short segment of various video clips from YouTube, focussing on the main actor and action depicted. The annotators had the option to watch the entire clip or only the segment to describe, with or without the audio, and could also choose the language for the description. In case English was not the native language of the annotator, the suggestion was given to use the Google translation service. These aspects made possible the collection of low-quality descriptions. Therefore, the authors organized the annotation process in two tasks to describe the same videos. Each annotator performed the first task. According to the quality of the English descriptions provided during the first task, the authors manually granted the best annotators with the access to the second task. Finally, however, the resulting dataset collected the descriptions from both the tasks:  $\sim 50k$  from the first task and  $\sim 30k$  from the second task.

Despite the instructions and the quality assessment procedure, the English portion of the MSVD contains syntactically and semantically incorrect descriptions. An example is reported in Fig. 5. Therefore, for this work, a task involving 21 users has been prepared, in which it has been asked to the users to check and correct all the captions of the MSVD. Note that simply removing the sentences with errors would have reduced the performance, as can be observed from TABLE III. For this reason, it has been preferred to amend the errors. Each caption correction has been double checked. For the task, four types of errors have been defined, and the annotators had to find and correct them. The types of errors, ranked based on their severity, are:

1. unsuitability, *i.e.*, the sentence has no meaning, is ill-formulated, or in general, does not respect the instructions

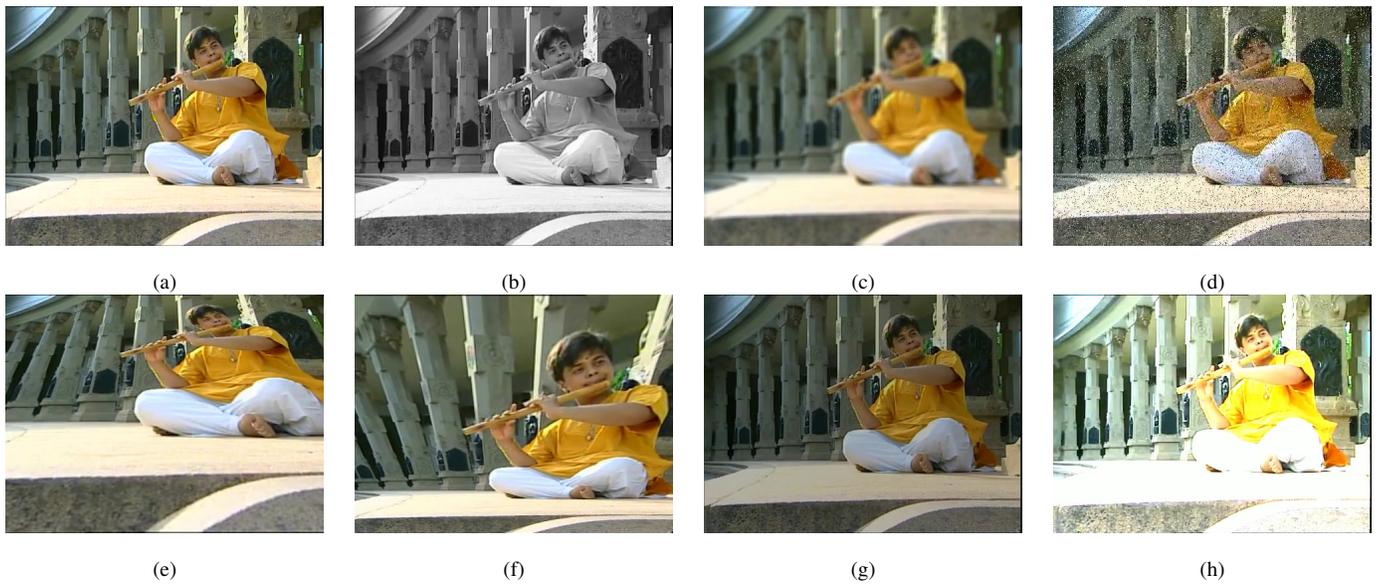


Fig. 3: Example of visual alterations applied to the training videos in the MSVD. 3a is an original image from the video. 3b is the grayscale converted image. 3c is the blurred image. 3d is the image with applied gaussian noise (salt and pepper noise). 3e and 3f are the image with two keystone distortions applied. 3g is the brightness reduced image. 3h is the brightness enhanced image.



Fig. 4: Example of visual style transfer applied to the training videos in the MSVD. 4a is an original image from the video. 4b is the image in the style of the Picasso's painting 'La Muse'. 4c is the image in the style of the Afremov's painting 'Rain Princess'. 4d is the image in the style of the Munch's painting 'The Scream'. 4e is the image in the style of the Picabia's painting 'Udnie'. 4f is the image in the style of the Hokusai's painting 'The Great Wave off Kanagawa'. 4g is the image in the style of the Turner's painting 'The Wreck of a Transport Ship'.

given in the original task of [68]. These sentences have been replaced with other correct descriptions of the same video.

2. hallucination, *i.e.*, the sentence describes actors or actions or objects that do not appear in the video. These errors have been corrected double-checking the video.
3. syntactic, *i.e.*, the sentence contains a grammatical error or a typo. These errors have been amended.
4. proper noun, *i.e.*, the sentence contains a proper noun, which cannot be inferred by the video but comes from the experience of the annotator. The proper nouns have been removed or replaced with a common one.

In the test subset, the annotators labelled the errors in



**Correct caption:** The man cried on the other man's shoulder.  
**Syntactic error:** A man **crys**.  
**Hallucination:** The person is **dancing**.  
**Proper noun:** **Prabhu Deva** is crying.  
**Unsuitability:** **Prabudeva song with kajol in merupukalalu**.

Fig. 5: Examples of captions with errors associated with a video in the MSVD.

Training captions per video	Training samples	$B_4$	$R_L$	$M$	$C$
1	1200	10.9	39.1	14.7	84.5
5	6000	29.8	56.7	24.1	<b>84.7</b>
10	12000	35.3	61.2	26.9	78.9
~43	49158	<b>45.1</b>	<b>69.4</b>	<b>32.9</b>	70.0

TABLE III: BEDDS model performance on the MSVD depending on the number of training captions per video.  $B_4$  stands for BLEU<sub>4</sub>,  $R_L$  for ROUGE<sub>L</sub>,  $M$  for METEOR, and  $C$  for CIDEr. Bold indicates the best performance.

	$B_4$	$R_L$	$M$	$C$
MSVD	60.10±6.49	77.52±5.27	43.61±3.59	124.30±20.54
MSVD-v2	<b>65.40±4.60</b>	<b>81.93±2.70</b>	<b>47.45±1.72</b>	<b>148.64±16.75</b>

TABLE IV: Human performance on the MSVD in its original version from [17], [68] and checked version from this work, MSVD-v2.  $B_4$  stands for BLEU<sub>4</sub>,  $R_L$  for ROUGE<sub>L</sub>,  $M$  for METEOR, and  $C$  for CIDEr. Bold indicates the best performance.

addition to correcting them. In case of multiple errors, the annotators labelled the caption giving priority to the most severe type of error. From this process, it emerged that the 24.62% of the captions in the test set contained one or more of the errors just described. The 49.20% of them had syntactical errors, 27.10% were unsuitable descriptions for the associated video, the 12.18% contained hallucinations, and the 11.52% proper nouns.

To gain insights into the MSVD, both the original one from [17], [68] and the one obtained for this study, referred to as MSVD-v2, the average human performance has been measured as follows. For each video, a ground truth sentence has been considered the predicted description, and the performance scores have been calculated similarly to what done for the automatic NLVD models. This procedure has been repeated 23 times since each video in the test subset of the MSVD is associated with at least 23 captions. Finally, the mean and standard deviation of the scores have been calculated. The human performance changes after checking the text part of the dataset as reported in TABLE IV. In particular, the mean value increases for all the scores and the variance decreases. This is no surprise considering the high number of detected and amended errors. In fact, the considered metrics are based on the similarity of words and groups of words in the compared sentences, and the dissimilarity due to the errors has been removed (or significantly reduced.) The values of the scores are not perfect because of the natural diversity in the possible ways to describe each video.

The obtained MSVD-v2 dataset is available online<sup>1</sup>.

#### IV. EXPERIMENTS AND RESULTS

In this section, the implementation details of the experimental setup used in this study are reported, and the obtained results are presented and discussed. The BEDDS model described in III-A has been used as the baseline for observing

the effects of the visual data augmentation and textual data cleansing preprocessing steps.

##### A. Implementation Details

1) *Architecture Details*: The dimension of the hidden state of the Encoder LSTM and GRU and the Decoder GRU has been set to 1000. When used, the learnt WE maps the words in a 300-dimensional space, and the VE maps the feature vectors in a 500-dimensional space. The vocabulary  $D$  has been built using the training and validation subsets of either the MSVD and MSVD-v2 datasets. In the first case, it consists of 10 160 words, in the second case, of 6428 words.

For the training, the Stochastic Gradient Descent algorithm has been employed, with learning rate set to 0.1 and kept constant. The batch size has been set to 64 samples. As the early stopping criterion, the METEOR score on the validation set has been used (similar to what done, e.g., in [48], [69].) In particular, the training has been stopped if the value of the METEOR score did not increase for ten consecutive epochs. In the test phase, the best model in terms of the METEOR score has been used. On average, the training ends in ~40 epochs for the models trained the original dataset, ~20 for the style augmented dataset, and ~15 for the classically augmented dataset. This resulted respectively in ~3h, ~24h, and ~48h for training the PyTorch implementation of the models on an NVIDIA Titan XP graphic card.

2) *Visual Data Augmentation Details*: Additional to the transformations described in V, in the test phase only it has been applied vertical flipping and contrast reduction and enhancement, with multiplicative factors 2 and 0.5 respectively. Apart from style transfer, vertical flipping and greyscale conversion, for all the applied transformation a parameter can be set to vary their severity. Different values of the parameters have been chosen for the transformations to the videos in the training set, and others for the transformations to the test set only. In particular:

- The kernel size  $\rho$  of the gaussian blur has been set to 12, 15, and 17 in training phase, and 5, 7, 10, and 20 in test phase only.
- The ratio between the top width  $w_{top}$  and the bottom width  $w_{bottom}$  of the image for the keystone distortion has been set to 5/2, 3, 2/5, and 1/3 in the training phase, and 3/2, 2, 2/3, and 1/2 in the test phase only.
- The enhancement and reduction factors for the brightness alteration have been set respectively to 2 and 0.2 in the training phase, and to 5, 7, and 0.5, 0.7 in the test phase only.
- The probability  $p$  that a pixel is affected by the Salt & Pepper noise has been set to 0.01, 0.05, and 0.1 in the training phase, and to 0.5, 0.7 in the test phase only.

##### B. Results

1) *Effects of Visual Data Augmentation*: The performance of the BEDDS model trained on the original training videos of the MSVD has been evaluated on the MSVD original test videos, and on the same videos altered as explained in IV-A2, to evaluate its generalization capability with respect

<sup>1</sup><http://sira.diei.unipg.it/supplementary/input4nlvd2018/>

to different visual conditions. The performance decreases proportionally with the intensity of the various transformations applied. This indicates that the model lacks robustness to unseen appearances of the scene.

The BEDDS model has been trained also on an augmented MSVD, obtained as explained in III-B and IV-A2. The resulting models are referred to as BEDDS-VA, in case of training on classically altered videos, and BEDDS-ST in case of training on style transformed videos. The comparison of the three models on the MSVD dataset is reported in TABLE V and in TABLE VI.

When tested on the original test videos of the MSVD, the performance the BEDDS-VA model is inferior to that of the BEDDS model trained on the original training videos only. However, on the test videos altered with the same transformations as in the training set, the BEDDS-VA model outperforms the BEDDS model in terms of all the metrics. Also for the BEDDS-VA model, the performance decreases proportionally with the intensity of the various transformations applied, but the performance drop is smaller than that of the BEDDS model trained on the original videos only. On test videos altered with unseen transformations, including style transfer, the BEDDS-VA model outperforms the BEDDS model in the majority of cases. This is particularly true for the performance in terms of the CIDEr metric, which is the one that by design better captures the human consensus on the quality of image descriptions. The cases in which the performance of the BEDDS model are comparable or superior to that of the BEDDS-VA model are those of transformations that do not significantly alter the appearance of the video, such as vertical flipping and small keystone distortion. This suggests that the BEDDS model is biased on the appearance of the training videos.

The BEDDS-ST model outperforms the BEDDS and BEDDS-VA models when tested on the style transformed videos and on severe Salt & Pepper noise alteration with probability of altered pixel set equals to 0.07. However, in the majority of the other cases, its performance is inferior to that of the other models. This suggests that, different from other Computer Vision tasks as classification under domain shift and depth estimation, performing style transfer for visual data augmentation is not effective for the NLVD task.

Some intuitions on this behaviour can be gained observing the data distribution obtained via the  $t$ -SNE analysis [70] depicted in Fig. 6. The points represent the *ResNet50* features extracted from the fifth frame of each video in the MSVD dataset, both original and altered as described in III-B. Recall that the *ResNet50* features capture the appearance of the frames. From the  $t$ -SNE plots, it can be observed that the style transformed frames form separate clusters, which do not overlap with the other data points. Some of the classically altered frames are grouped together, *e.g.*, those altered via Salt & Pepper noise, severe Gaussian blur, and brightness variation. In such cases, the BEDDS-VA model outperforms the BEDDS model. The transformations that do not severely alter the appearance of the videos result in points that are distributed as those corresponding to the original frames. Therefore, in such cases, the BEDDS model performs comparably or better

than the BEDDS-VA model. In applicative scenarios, the same analysis can be performed on videos captured under the specific operating conditions. This can guide the selection of the most appropriate visual transformations to apply to videos to include in the dataset for training or finetuning the NLVD system.

Furthermore, the performance of the BEDDS, BEDDS-VA, and BEDDS-ST models have been tested without retraining on the MSR-VTT dataset. This dataset has characteristics similar to those of the MSVD dataset, in terms of visual quality of the videos and number and quality of the captions, since both contain videos from YouTube with multiple captions per videos, collected via the AMT service. The results of this comparison are reported in TABLE VII. The performance of the three models are comparable, and all below the human performance on the same dataset, calculated as done for the MSVD dataset in III-C.

These results suggest that with the visual data augmentation preprocessing step the model can deal better with appearance changes. However, the recently proposed style augmentation approach results less effective than classical alterations in the context of NLVD. In addition, the robustness with respect to appearance conditions of specific applications can be further increased by training the NLVD models on videos altered accordingly.

2) *Effects of Textual Data Cleansing*: The BEDDS model has been trained on either the MSVD and the MSVD-v2 dataset, obtained as explained in III-C. The resulting model is referred to as BEDDS-TC. Both the variants have been tested on the two datasets. The BEDDS-TC model outperforms the BEDDS model on both datasets in terms of all metrics but CIDEr on the MSVD (69.8 for BEDDS-TC and 70.0 for BEDDS.) The results of this study are reported in TABLE VIII. The same trend can be observed also when testing on the MSR-VTT dataset, as observed from TABLE VII.

Considering only the performance gain obtained in terms of evaluation metrics is limiting and can be misleading for investigating the effects of training with high-quality textual data. Therefore, the descriptions produced by the two models have been compared further, from a qualitative point of view. The complete corpus of results is available online<sup>2</sup>. As expected, there are cases where one model outputs a correct description while the other a completely wrong one. Nevertheless, both the BEDDS-TC and the BEDDS models produce correct detailed descriptions for the same videos. It is interesting to focus on the cases where the BEDDS model outputs an erroneous detailed description. Some examples are reported in Fig. 7 for the MSVD dataset, and in Fig. 8 for the MSR-VTT dataset. In such cases, the descriptions from the BEDDS-TC model are more generic but still correct. However, metrics based on  $n$ -gram similarity rather than semantic consistency, like those used in the NLVD evaluation, cannot properly capture this aspect. In addition, synonyms and hypernyms can be penalized [71]. This can explain the little performance gain achieved with textual data cleansing in terms of such metrics. In the sight of this and of the considerations in III-C on the syntactic and se-

<sup>2</sup><http://sira.diei.unipg.it/supplementary/input4nlvd2018/>

		BEDDS				BEDDS-VA				BEDDS-ST			
<i>Alteration</i>		$B_4$	$R_L$	$M$	$C$	$B_4$	$R_L$	$M$	$C$	$B_4$	$R_L$	$M$	$C$
In BEDDS-VA Training and Test Phase	None ( <i>i.e.</i> , Original videos)	<b>45.1</b>	<b>69.4</b>	<b>32.9</b>	<b>70.0</b>	43.5	69.2	32.6	69.7	42.5	68.1	31.6	62.7
	Greyscale Conversion	40.4	66.6	30.7	57.8	<b>43.5</b>	<b>69.1</b>	<b>32.0</b>	<b>66.3</b>	40.1	66.9	30.4	59.1
	Gaussian Blur with $\rho = 12$	39.2	65.6	29.6	50.6	<b>41.4</b>	<b>68.6</b>	<b>30.9</b>	<b>64.0</b>	37.7	64.8	28.5	46.8
	Gaussian Blur with $\rho = 15$	36.4	64.5	28.3	44.3	<b>40.5</b>	<b>66.6</b>	<b>30.4</b>	<b>59.6</b>	35.0	63.4	27.3	41.1
	Gaussian Blur with $\rho = 17$	34.8	63.3	27.4	39.3	<b>43.2</b>	<b>68.6</b>	<b>31.7</b>	<b>67.1</b>	33.3	62.7	26.7	37.6
	Keystone Distortion $w_{top}/w_{bottom} = 2/5$	40.3	67.2	30.9	59.1	<b>43.2</b>	<b>68.6</b>	<b>31.7</b>	<b>70.2</b>	36.0	64.0	28.3	47.5
	Keystone Distortion $w_{top}/w_{bottom} = 1/3$	38.9	66.6	30.4	56.0	<b>40.0</b>	<b>67.7</b>	<b>30.8</b>	<b>61.9</b>	34.1	63.3	27.1	40.4
	Keystone Distortion $w_{top}/w_{bottom} = 5/2$	39.1	66.7	30.2	59.5	<b>41.5</b>	<b>67.6</b>	<b>31.1</b>	<b>66.3</b>	37.3	64.5	28.3	50.0
	Keystone Distortion $w_{top}/w_{bottom} = 3$	36.3	65.3	28.8	48.8	<b>40.7</b>	<b>66.9</b>	<b>29.4</b>	<b>59.7</b>	34.3	62.7	26.7	43.7
	Brightness Reduction $\times 0.2$	38.7	64.9	29.0	53.8	<b>42.1</b>	<b>68.3</b>	<b>31.2</b>	<b>64.6</b>	38.0	65.0	28.8	53.9
	Brightness Enhancement $\times 2$	39.2	67.1	30.5	57.3	<b>42.0</b>	<b>67.8</b>	<b>31.2</b>	<b>64.0</b>	37.3	65.3	29.2	54.0
	Salt & Pepper noise with $p = 0.01$	26.5	59.9	24.3	36.3	<b>40.4</b>	<b>66.6</b>	<b>31.0</b>	<b>62.6</b>	33.7	62.8	27.1	43.2
	Salt & Pepper noise with $p = 0.05$	22.7	58.7	23.1	26.4	<b>39.2</b>	<b>65.9</b>	<b>30.0</b>	<b>54.6</b>	30.5	61.5	25.6	32.8
	Salt & Pepper noise with $p = 0.1$	22.9	58.7	23.1	23.8	<b>38.0</b>	<b>65.4</b>	<b>29.4</b>	<b>53.2</b>	28.2	59.9	24.5	29.3
In Test Phase Only	Vertical Flipping	<b>44.5</b>	<b>69.4</b>	<b>32.9</b>	<b>70.3</b>	43.4	69.1	32.3	68.3	40.6	67.2	30.8	58.6
	Gaussian Blur with $\rho = 5$	<b>44.4</b>	69.0	32.2	68.5	43.6	<b>69.1</b>	<b>32.4</b>	<b>69.9</b>	41.5	67.6	30.8	58.6
	Gaussian Blur with $\rho = 7$	<b>43.2</b>	68.1	31.5	62.5	42.7	<b>68.7</b>	<b>32.0</b>	<b>67.8</b>	40.3	67.0	30.6	56.2
	Gaussian Blur with $\rho = 10$	41.4	67.0	30.7	57.3	<b>42.5</b>	<b>68.3</b>	<b>31.9</b>	<b>66.5</b>	39.6	65.9	29.4	51.7
	Gaussian Blur with $\rho = 20$	32.2	61.7	26.2	32.9	<b>38.4</b>	<b>65.4</b>	<b>29.3</b>	<b>54.9</b>	31.7	61.8	25.8	32.0
	Keystone Distortion $w_{top}/w_{bottom} = 2/3$	<b>44.3</b>	69.1	<b>32.6</b>	67.8	43.6	<b>69.4</b>	32.3	<b>70.8</b>	42.7	67.8	31.6	62.0
	Keystone Distortion $w_{top}/w_{bottom} = 1/2$	<b>42.8</b>	<b>68.5</b>	<b>31.7</b>	64.5	40.9	67.8	31.6	<b>71.5</b>	39.1	66.0	29.9	53.9
	Keystone Distortion $w_{top}/w_{bottom} = 3/2$	<b>45.4</b>	<b>69.5</b>	<b>32.6</b>	<b>69.8</b>	43.5	<b>69.5</b>	32.5	<b>69.8</b>	42.4	67.9	31.5	64.9
	Keystone Distortion $w_{top}/w_{bottom} = 2$	41.4	67.9	31.3	64.3	<b>43.1</b>	<b>68.7</b>	<b>32.1</b>	<b>69.8</b>	39.7	65.8	29.6	57.7
	Brightness Reduction $\times 0.5$	<b>44.2</b>	<b>69.2</b>	32.3	68.2	43.4	69.1	<b>32.5</b>	<b>70.7</b>	41.4	67.8	31.0	61.8
	Brightness Reduction $\times 0.7$	45.1	<b>69.4</b>	<b>32.8</b>	71.6	<b>46.6</b>	69.2	32.6	<b>74.2</b>	41.4	67.9	31.2	61.6
	Brightness Enhancement $\times 5$	24.0	57.7	23.1	29.1	<b>27.1</b>	<b>59.2</b>	<b>24.4</b>	<b>33.3</b>	25.2	57.4	23.2	26.8
	Brightness Enhancement $\times 7$	19.4	<b>55.2</b>	21.2	18.2	<b>21.5</b>	<b>55.2</b>	<b>21.8</b>	<b>21.2</b>	18.9	53.4	20.2	16.9
	Salt & Pepper noise with $p = 0.5$	10.0	<b>52.7</b>	18.6	2.8	<b>15.4</b>	52.2	17.8	<b>7.8</b>	12.8	52.6	<b>19.0</b>	3.2
	Salt & Pepper noise with $p = 0.7$	8.4	51.9	18.6	1.8	9.7	49.0	14.7	2.0	<b>10.4</b>	<b>53.9</b>	<b>21.0</b>	<b>2.1</b>
	Contrast Reduction $\times 0.5$	<b>44.0</b>	68.5	31.8	64.7	42.8	<b>69.0</b>	<b>31.9</b>	<b>68.1</b>	42.3	67.8	31.2	62.4
	Contrast Enhancement $\times 2$	41.5	67.5	30.9	60.9	<b>41.9</b>	<b>68.6</b>	<b>31.6</b>	<b>65.6</b>	38.7	66.1	29.7	57.7

TABLE V: Performance of the BEDDS, BEDDS-VA, and BEDDS-ST models on differently altered test videos of the MSVD, used both in training and test phase or in test phase only.  $B_4$  stands for BLEU<sub>4</sub>,  $R_L$  for ROUGE<sub>L</sub>,  $M$  for METEOR, and  $C$  for CIDEr. Bold indicates the best performance.

mantic errors in the MSVD, we believe that using the MSVD-v2 dataset to train and test the NLVD algorithms is reasonable because it contains better quality ground truth captions. This is confirmed by the average human performance estimation on the MSVD and MSVD-v2 datasets. As mentioned in III-C, its mean value is higher on the amended dataset, and the variance is smaller. Neither human performance can be perfect for this task, due to its intrinsic subjectivity. However, the improved performance after the textual data cleansing suggests that the MSVD-v2 dataset represents a more reliable benchmark than the MSVD for the NLVD task. Finally, the comparison of the performance on original and the amended datasets highlights the importance of the consistency of the textual component when designing an NLVD system.

## V. CONCLUSION

In this work, it has been presented a study to evaluate the performance of NLVD systems in case the video input dataset is augmented with transformed video derived from the original ones applying common transformations. For this purpose, extensive studies have been performed on the benchmark MSVD dataset and on a refined version specifically amended

for this study (the MSVD-v2 dataset.) The experiments have been carried out using a simple yet effective NLVD encoder-decoder architecture.

The results of the analysis reveal that the visual data augmentation generally provides improvements in terms of robustness to appearance changes. In particular, considering the CIDEr score, which by design correlates with the human judgment on image description, the model trained on the augmented videos obtains an average +4.5% performance improvement with peaks up to +22.0% for severe Gaussian blur, when tested on videos altered using a different set of transformations compared to those used in the training set. As expected, this improvement is more significant when the NLVD model is tested on videos altered with the same transformation used in the training set (+12.7% on average, with peaks up to +29,4% for severe alterations as keystone distortion and Salt & Pepper noise.) This suggests that, when applying the NLVD system in a real-world scenario, it is beneficial to train or finetune the system with videos altered according to the visual conditions typical of the specific application. In this work, it has been shown that some insights on the utility of the specific input transformations can be

Style	BEDDS				BEDDS-VA				BEDDS-ST			
	$B_4$	$R_L$	$M$	$C$	$B_4$	$R_L$	$M$	$C$	$B_4$	$R_L$	$M$	$C$
Original videos	<b>45.1</b>	<b>69.4</b>	<b>32.9</b>	<b>70.0</b>	43.5	69.2	32.6	69.7	42.5	68.1	31.6	62.7
Picasso's La Muse	10.3	43.4	16.4	7.6	13.6	51.3	19.2	7.2	<b>31.9</b>	<b>61.8</b>	<b>26.5</b>	<b>38.9</b>
Afremov's Rain Princess	18.5	51.1	19.9	20.6	13.9	48.2	19.2	12.4	<b>34.8</b>	<b>63.3</b>	<b>26.8</b>	<b>44.0</b>
Munch's The Scream	27.9	60.5	25.5	36.9	27.8	60.5	25.4	38.5	<b>38.9</b>	<b>65.8</b>	<b>29.6</b>	<b>54.4</b>
Picabia's Udnie	23.0	57.2	23.1	21.6	23.0	57.8	23.2	21.4	<b>35.5</b>	<b>64.2</b>	<b>27.9</b>	<b>48.0</b>
Hokusai's The Great Wave off Kanagawa	22.9	57.9	22.8	24.6	24.2	58.4	24.1	26.9	<b>38.8</b>	<b>65.5</b>	<b>29.1</b>	<b>50.8</b>
Turner's The Wreck of a Transport Ship	25.4	59.3	23.6	30.2	26.6	60.5	24.8	34.8	<b>38.2</b>	<b>65.3</b>	<b>28.9</b>	<b>52.8</b>

TABLE VI: Performance of the BEDDS, BEDDS-VA, and BEDDS-ST models on the test videos of the MSVD, transformed in different artistic styles.  $B_4$  stands for BLEU<sub>4</sub>,  $R_L$  for ROUGE<sub>L</sub>,  $M$  for METEOR, and  $C$  for CIDEr. Bold indicates the best performance.

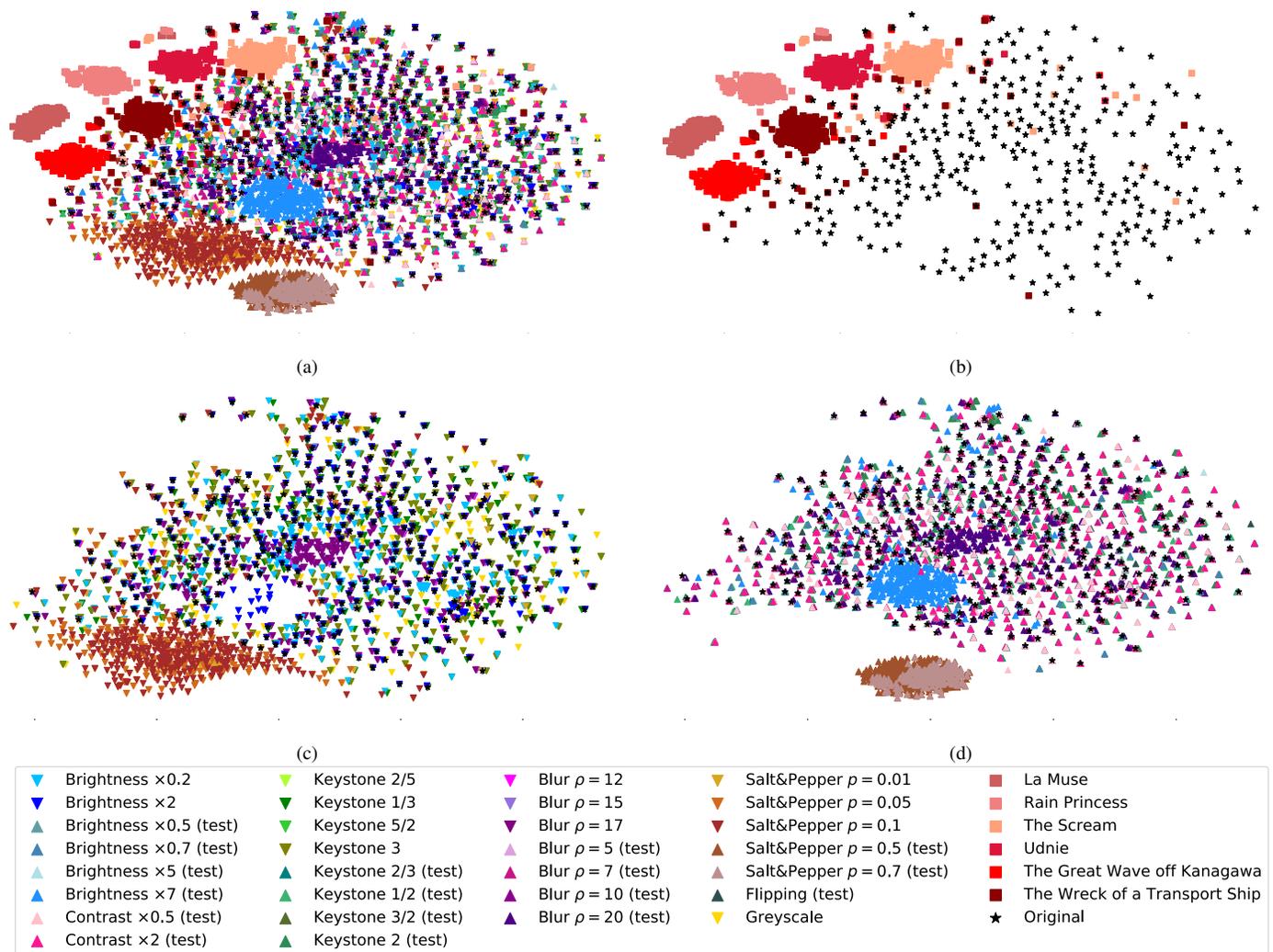


Fig. 6: Distribution of the frames in the MSVD dataset, altered with classical alterations and style transfer. 6a contains the points associated to the original frames and to all the altered frames. 6b contains the points associated to the original frames and to the style transformed frames used for training the BEDDS-ST model. 6c contains the points associated to the original frames and to the frames altered with the classical alterations in the training set of the BEDDS-VA model. 6d contains the points associated to the original frames and to the altered frames used only for test.

	$B_4$	$R_L$	$M$	$C$
BEDDS	16.9	42.7	16.3	<b>9.6</b>
BEDDS-VA	16.9	42.2	16.0	9.0
BEDDS-ST	16.9	42.1	16.0	8.6
BEDDS-TC	<b>17.5</b>	<b>43.0</b>	<b>16.5</b>	9.3
Humans	23.4 ± 3.6	44.7 ± 1.2	23.5 ± 0.7	31.2 ± 1.6

TABLE VII: Performance of the BEDDS, BEDDS-VA, and BEDDS-ST models on the original test videos of the MSR-VTT.  $B_4$  stands for BLEU<sub>4</sub>,  $R_L$  for ROUGE<sub>L</sub>,  $M$  for METEOR, and  $C$  for CIDEr. Bold indicates the best performance of the models. For completeness, the human performance are also reported.

		$B_4$	$R_L$	$M$	$C$
MSVD	BEDDS	45.1	69.4	32.9	<b>70.0</b>
	BEDDS-TC	<b>45.8</b>	<b>70.1</b>	<b>33.1</b>	69.8
MSVD-v2	BEDDS	44.6	69.2	32.6	68.7
	BEDDS-TC	<b>45.5</b>	<b>70.0</b>	<b>33.1</b>	<b>79.5</b>

TABLE VIII: Performance of the BEDDS and BEDDS-TC models on the two versions of the MSVD, original and checked (MSVD-v2).  $B_4$  stands for BLEU<sub>4</sub>,  $R_L$  for ROUGE<sub>L</sub>,  $M$  for METEOR, and  $C$  for CIDEr. Bold indicates the best performance.

gained using a t-SNE analysis. Specifically, the videos altered via transformations that do not severely change the appearance are distributed as the original videos, while those altered with severe transformations (such as Salt & Pepper noise, severe Gaussian blur and brightness variation) are grouped in separate clusters. For those latter cases, data augmentation brings to a significant improvement in the performance of the NLVD system. Finally, it was observed that the BEDDS-TC model, trained on the refined MSVD-v2 dataset, provides more generic but correct captions, reflected in a performance improvement in terms of all the evaluation metrics.

#### ACKNOWLEDGMENT

We gratefully thank the NVIDIA Corporation with the donation of the *Titan XP* GPU used for this research. Our gratitude also goes to the users who volunteered for the MSVD text checking task.

#### REFERENCES

- [1] P. Cui, S. Liu, and W. Zhu, "General knowledge embedded image representation learning," *IEEE Transactions on Multimedia*, vol. 20, no. 1, pp. 198–207, 2018.
- [2] C. Kofler, L. Yang, M. Larson, T. Mei, A. Hanjalic, and S. Li, "Predicting failing queries in video search," *IEEE Transactions on Multimedia*, vol. 16, no. 7, pp. 1973–1985, 2014.
- [3] H. Xie, Y. Zhang, J. Tan, L. Guo, and J. Li, "Contextual query expansion for image retrieval," *IEEE Transactions on Multimedia*, vol. 16, no. 4, pp. 1104–1114, 2014.
- [4] W. Li, J. Joo, H. Qi, and S.-C. Zhu, "Joint image-text news topic detection and tracking by multimodal topic and-or graph," *IEEE Trans. Multimedia*, vol. 19, no. 2, pp. 367–381, 2017.
- [5] Y. Hu, L. Zheng, Y. Yang, and Y. Huang, "Twitter100k: A real-world dataset for weakly supervised cross-media retrieval," *IEEE Transactions on Multimedia*, vol. 20, no. 4, pp. 927–938, 2018.



**Reference:** A woman adds starch powder and a little water to two shrimps in a bowl, mixes them to coat evenly on both sides.

**BEDDS:** A woman is mixing a mixture of meat.

**BEDDS-TC:** A woman is cooking.



**Reference:** A short haired furry animal is holding something in its paws and nibbling on it.

**BEDDS:** A cat is playing with a toy.

**BEDDS-TC:** A small animal is biting a stuffed.



**Reference:** A man sprinkles salt and various spices on pieces of chicken placed in a bowl and tosses them gently.

**BEDDS:** A man is cutting meat.

**BEDDS-TC:** A person is cooking.

Fig. 7: Exemplar captions produced by the BEDDS model, which was trained on the original MSVD, and the BEDDS-TC model, which was trained on the MSVD-v2 dataset.

- [6] H. Song, X. Wu, W. Yu, and Y. Jia, "Extracting key segments of videos for event detection by learning from web sources," *IEEE Transactions on Multimedia*, vol. 20, no. 5, pp. 1088–1100, 2018.
- [7] X. Yang, T. Zhang, and C. Xu, "Text2video: An end-to-end learning framework for expressing text with videos," *IEEE Transactions on Multimedia*, 2018.
- [8] L. Baraldi, C. Grana, and R. Cucchiara, "Recognizing and presenting the storytelling video structure with deep multimodal networks," *IEEE Transactions on Multimedia*, vol. 19, no. 5, pp. 955–968, 2017.
- [9] L. Li, S. Tang, Y. Zhang, L. Deng, and Q. Tian, "Gla: Global-local attention for image description," *IEEE Transactions on Multimedia*, vol. 20, no. 3, pp. 726–737, 2018.
- [10] L. Gao, Z. Guo, H. Zhang, X. Xu, and H. T. Shen, "Video captioning with attention-based lstm and semantic consistency," *IEEE Transactions on Multimedia*, vol. 19, no. 9, pp. 2045–2055, 2017.
- [11] J. Dong, X. Li, and C. G. Snoek, "Predicting visual features from text for image and video caption retrieval," *IEEE Transactions on Multimedia*, 2018.
- [12] R. Vedantam, C. Lawrence Zitnick, and D. Parikh, "CIDEr: Consensus-based Image Description Evaluation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 4566–4575.
- [13] P. Anderson, B. Fernando, M. Johnson, and S. Gould, "Spice: Semantic propositional image caption evaluation," in *European Conference on Computer Vision*. Springer, 2016, pp. 382–398.
- [14] S. Cascianelli, G. Costante, T. A. Ciarfuglia, P. Valigi, and M. L. Fravolini, "Full-gru natural language video description for service



**Reference:** A woman in a black shirt and a woman in a purple shirt play badminton.

**BEDDS:** A girl is dancing.

**BEDDS-TC:** A girl is playing a game.



**Reference:** A man and woman are talking to each other on a news show

**BEDDS:** A woman is speaking on a stage.

**BEDDS-TC:** A woman is talking.



**Reference:** Some cookies in a plate are served on table and a chef is talking.

**BEDDS:** A man is eating a meal.

**BEDDS-TC:** A man is talking about a table.

Fig. 8: Exemplar captions produced by the BEDDS model and the BEDDS-TC model on the MSR-VTT dataset.

robotics applications,” *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 841–848, 2018.

- [15] A. Torabi, C. Pal, H. Larochelle, and A. Courville, “Using descriptive video services to create a large data source for video annotation research,” *arXiv preprint arXiv:1503.01070*, 2015.
- [16] A. Rohrbach, M. Rohrbach, N. Tandon, and B. Schiele, “A dataset for movie description,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3202–3212.
- [17] S. Guadarrama, N. Krishnamoorthy, G. Malkarnenkar, S. Venugopalan, R. Mooney, T. Darrell, and K. Saenko, “Youtube2text: Recognizing and describing arbitrary activities using semantic hierarchies and zero-shot recognition,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 2712–2719.
- [18] J. Xu, T. Mei, T. Yao, and Y. Rui, “Msr-vtt: A large video description dataset for bridging video and language,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 5288–5296.
- [19] A. Rohrbach, A. Torabi, M. Rohrbach, N. Tandon, C. Pal, H. Larochelle, A. Courville, and B. Schiele, “Movie description,” *International Journal of Computer Vision*, vol. 123, no. 1, pp. 94–120, 2017.
- [20] C.-M. Teng, “Correcting noisy data.” in *ICML*. Citeseer, 1999, pp. 239–248.
- [21] S. Kotsiantis, D. Kanellopoulos, and P. Pintelas, “Data preprocessing for supervised learning,” *International Journal of Computer Science*, vol. 1, no. 2, pp. 111–117, 2006.
- [22] N. M. Nawi, W. H. Atomi, and M. Rehman, “The effect of data preprocessing on optimized training of artificial neural networks,” *Procedia Technology*, vol. 11, pp. 32–39, 2013.
- [23] J. Thomason, S. Venugopalan, S. Guadarrama, K. Saenko, and R. J. Mooney, “Integrating language and vision to generate natural language descriptions of videos in the wild.” in *Coling*, vol. 2, no. 5, 2014, p. 9.
- [24] K. Cho, A. Courville, and Y. Bengio, “Describing multimedia content using attention-based encoder-decoder networks,” *IEEE Transactions on Multimedia*, vol. 17, no. 11, pp. 1875–1886, 2015.
- [25] N. Krishnamoorthy, G. Malkarnenkar, R. J. Mooney, K. Saenko, and S. Guadarrama, “Generating Natural-Language Video Descriptions Using Text-Mined Knowledge.” in *AAAI*, vol. 1, 2013, p. 2.
- [26] Y. Zhu, R. Kiros, R. Zemel, R. Salakhutdinov, R. Urtasun, A. Torralba, and S. Fidler, “Aligning Books and Movies: Towards Story-like Visual Explanations by Watching Movies and Reading Books,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 19–27.
- [27] S. Venugopalan, M. Rohrbach, J. Donahue, R. Mooney, T. Darrell, and K. Saenko, “Sequence to Sequence-Video to Text,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 4534–4542.
- [28] Y. Guo, J. Zhang, and L. Gao, “Exploiting long-term temporal dynamics for video captioning,” *World Wide Web*, pp. 1–15, 2018.
- [29] Y. Xu, Y. Han, R. Hong, and Q. Tian, “Sequential video vlad: training the aggregation locally and temporally,” *IEEE Transactions on Image Processing*, vol. 27, no. 10, pp. 4933–4944, 2018.
- [30] S. Venugopalan, H. Xu, J. Donahue, M. Rohrbach, R. Mooney, and K. Saenko, “Translating Videos to Natural Language using Deep Recurrent Neural Networks,” *arXiv preprint arXiv:1412.4729*, 2014.
- [31] X. Li, Z. Zhou, L. Chen, and L. Gao, “Residual attention-based lstm for video captioning,” *World Wide Web*, pp. 1–16, 2018.
- [32] Y. Bin, Y. Yang, F. Shen, N. Xie, H. T. Shen, and X. Li, “Describing video with attention-based bidirectional lstm,” *IEEE Transactions on Cybernetics*, 2018.
- [33] L. Baraldi, C. Grana, and R. Cucchiara, “Hierarchical boundary-aware neural encoder for video captioning,” in *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*. IEEE, 2017, pp. 3185–3194.
- [34] T. Mikolov, K. Chen, G. Corrado, and J. Dean, “Efficient estimation of word representations in vector space,” *arXiv preprint arXiv:1301.3781*, 2013.
- [35] J. Pennington, R. Socher, and C. Manning, “Glove: Global vectors for word representation,” in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 1532–1543.
- [36] J. Song, Y. Guo, L. Gao, X. Li, A. Hanjalic, and H. T. Shen, “From deterministic to generative: Multi-modal stochastic rnns for video captioning,” *arXiv preprint arXiv:1708.02478*, 2017.
- [37] S. Venugopalan, L. A. Hendricks, R. Mooney, and K. Saenko, “Improving LSTM-based Video Description with Linguistic Knowledge Mined from Text,” *arXiv preprint arXiv:1604.01729*, 2016.
- [38] H. Wang, C. Gao, and Y. Han, “Sequence in sequence for video captioning,” *Pattern Recognition Letters*, 2018. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0167865518303234>
- [39] W. Li, D. Guo, and X. Fang, “Multimodal architecture for video captioning with memory networks and an attention mechanism,” *Pattern Recognition Letters*, vol. 105, pp. 23–29, 2018.
- [40] T.-H. Chen, K.-H. Zeng, W.-T. Hsu, and M. Sun, “Video captioning via sentence augmentation and spatio-temporal attention,” in *Asian Conference on Computer Vision*. Springer, 2016, pp. 269–286.
- [41] Z. Yang, Y. Han, and Z. Wang, “Catching the temporal regions-of-interest for video captioning,” in *Proceedings of the 2017 ACM on Multimedia Conference*. ACM, 2017, pp. 146–153.
- [42] A. Wu and Y. Han, “Multi-modal circulant fusion for video-to-language and backward,” in *IJCAI*, 2018, pp. 1029–1035.
- [43] Y. Yu, J. Kim, and G. Kim, “A joint sequence fusion model for video question answering and retrieval,” in *The European Conference on Computer Vision (ECCV)*, September 2018.
- [44] R. Pasunuru and M. Bansal, “Multi-task video captioning with video and entailment generation,” *arXiv preprint arXiv:1704.07489*, 2017.
- [45] L. Li and B. Gong, “End-to-end video captioning with multitask reinforcement learning,” *arXiv preprint arXiv:1803.07950*, 2018.
- [46] R. Pasunuru and M. Bansal, “Reinforced video captioning with entailment rewards,” *arXiv preprint arXiv:1708.02300*, 2017.
- [47] X. Wang, W. Chen, J. Wu, Y.-F. Wang, and W. Y. Wang, “Video captioning via hierarchical reinforcement learning,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4213–4222.
- [48] B. Wang, L. Ma, W. Zhang, and W. Liu, “Reconstruction network for video captioning,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7622–7631.

- [49] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Advances in Neural Information Processing Systems*, 2012, pp. 1097–1105.
- [50] P. T. Jackson, A. Atapour-Abarghouei, S. Bonner, T. Breckon, and B. Obara, "Style augmentation: Data augmentation via style randomization," *arXiv preprint arXiv:1809.05375*, 2018.
- [51] G. Ghiasi, H. Lee, M. Kudlur, V. Dumoulin, and J. Shlens, "Exploring the structure of a real-time, arbitrary neural artistic stylization network," *arXiv preprint arXiv:1705.06830*, 2017.
- [52] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2414–2423.
- [53] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *European Conference on Computer Vision*. Springer, 2016, pp. 694–711.
- [54] V. L. D. U. A. Vedaldi, "Instance normalization: The missing ingredient for fast stylization," *arXiv preprint arXiv:1607.08022*, 2016.
- [55] Y. Li, M.-Y. Liu, X. Li, M.-H. Yang, and J. Kautz, "A closed-form solution to photorealistic image stylization," *arXiv preprint arXiv:1802.06474*, 2018.
- [56] X. Zhang and Y. LeCun, "Text understanding from scratch," *arXiv preprint arXiv:1502.01710*, 2015.
- [57] C. Fellbaum, "Wordnet," in *Theory and applications of ontology: computer applications*. Springer, 2010, pp. 231–243.
- [58] I. Saito, J. Suzuki, K. Nishida, K. Sadamitsu, S. Kobashikawa, R. Masumura, Y. Matsumoto, and J. Tomita, "Improving neural text normalization with data augmentation at character-and morphological levels," in *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, vol. 2, 2017, pp. 257–262.
- [59] M. Fadaee, A. Bisazza, and C. Monz, "Data augmentation for low-resource neural machine translation," *arXiv preprint arXiv:1705.00440*, 2017.
- [60] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "BLEU: a Method for Automatic Evaluation of Machine Translation," in *Proceedings of the 40th annual meeting on Association for Computational Linguistics*. Association for Computational Linguistics, 2002, pp. 311–318.
- [61] C.-Y. Lin, "ROUGE: A Package for Automatic Evaluation of Summaries," in *Text summarization branches out: Proceedings of the ACL-04 workshop*, vol. 8. Barcelona, Spain, 2004.
- [62] S. Banerjee and A. Lavie, "METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments," in *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, vol. 29, 2005, pp. 65–72.
- [63] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.
- [64] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning Spatiotemporal Features with 3D Convolutional Networks," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 4489–4497.
- [65] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [66] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling," *arXiv preprint arXiv:1412.3555*, 2014.
- [67] L. Engstrom, "Fast style transfer," <https://github.com/lengstrom/fast-style-transfer/>, 2016, commit c77c028.
- [68] D. L. Chen and W. B. Dolan, "Collecting highly parallel data for paraphrase evaluation," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (ACL-2011)*, Portland, OR, June 2011.
- [69] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *International conference on machine learning*, 2015, pp. 2048–2057.
- [70] L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of machine learning research*, vol. 9, no. Nov, pp. 2579–2605, 2008.
- [71] M. Kilickaya, A. Erdem, N. Ikizler-Cinbis, and E. Erdem, "Re-evaluating automatic metrics for image captioning," *arXiv preprint arXiv:1612.07600*, 2016.
- Silvia Cascianelli** received the M.Sc. *magna cum laude* degree in Information and Automation Engineering in 2015. She then joined the Intelligent Systems, Automation and Robotics Laboratory (ISARLab) in 2015 as a Ph.D. student and she is currently a Research Assistant there. Her research interests are mainly Machine Learning, Natural Language Processing, and Computer Vision for Robotics.
- Gabriele Costante** received the Ph.D. degree in Robotics from the University of Perugia in 2016. He is currently a Post-Doc Researcher at the ISARLab and a Lecturer of Computer Vision at the University of Perugia, Department of Engineering. His research interests are mainly Robotics, Computer Vision and Machine Learning.
- Alessandro Devo** received the M.Sc. *magna cum laude* degree in Information and Robotics Engineering in 2018 from University of Perugia, with a thesis on Natural Language Video Description for Service Robotics Applications from the University of Perugia. He then joined the ISARLab as a Ph.D. Student. His research interests are mainly Machine Learning, Reinforcement Learning, and Computer Vision.
- Thomas A. Ciarfuglia** received the Ph.D. degree in Robotics from the University of Perugia in 2011. He joined the ISARLab in 2008 and worked as a Post-Doc there. He is a Lecturer of Machine Learning at the University of Perugia, Department of Engineering. His research interests are Machine Learning and Computer Vision for Robotics.
- Paolo Valigi** received the Ph.D. degree from University of Rome "Tor Vergata" in 1991. From 1990 to 1994 he worked with the Fondazione Ugo Bordoni. Since 2004 he has been Full Professor at the University of Perugia, Department of Engineering. He is currently the head of the ISARLab. His research interests are in the field of Robotics and Systems Biology.
- Mario L. Fravolini** received the Ph.D. degree in Electronic Engineering from the University of Perugia in 2000. He worked as a Research Assistant in the Control Group at the School of Aerospace Engineering, Georgia Institute of Technology, and at the Department of Mechanical and Aerospace Engineering West Virginia University. Currently, he is an Associate Professor at the Department of Engineering, University of Perugia. His research interests include: Fault Diagnosis, Intelligent and Adaptive Control and Biomedical Imaging.