Video-Text Compliance: Activity Verification Based on Natural Language Instructions

Mayoore S. Jaiswal[1*] Frank Liu[2*†] Anupama Jagannathan[1] Anne Gattiker[1] Inseok Hwang[1]
Jinho Lee[3†] Matt Tong[4] Sahil Dureja[4] Soham Shah[5†] Peter Hofstee[1]
Valerie Chen[6 ‡] Suvadip Paul[7 ‡] Rogerio Feris[4]
[1]IBM [2]Oak Ridge National Laboratory [3]Yonsei University [4]IBM Research
[5]Brain Technologies, Inc. [6]Yale University [7]Stanford University
mayoore.s.jaiswal@ibm.com

Abstract

We define a new multi-modal compliance problem, which is to determine if the human activity in a given video is in compliance with an associated text instruction. Learning at the junction of vision and text for the compliance problem requires addressing the challenges caused by irregularities in videos and ambiguities in natual language. Successful solutions to the compliance problem could enable automatic compliance checking and efficient feedback in many real-world settings. To this end, we introduce the Video-Text Compliance (VTC) dataset, which contains videos of atomic activities, along with text instructions and compliance labels. The VTC dataset is constructed by an auto-augmentation technique, preserves privacy, and contains over 1.2 *million frames. Finally we present ComplianceNet, a novel end-to-end trainable network to solve the video-text compliance task. Trained on the VTC dataset, ComplianceNet improves the baseline accuracy by* 27.5% *on average. We plan to release the VTC dataset to the community for future research.*

1. Introduction

Technology advances have made video recording devices pervasive. Almost every one of the 1.5 billion smartphones sold in 2018 [5] is capable of recording videos. Effectively utilizing the vast quantity of video data, rather than simply storing it, can open many opportunities to improve peoples' lives. To address this challenge, one of the active research directions in the computer vision community is the joint learning of visual and textual data, with examples in action recognition [40, 10, 47], temporal action proposal [16, 23, 27], and alignment of video frames and language descriptions [29, 6, 23].

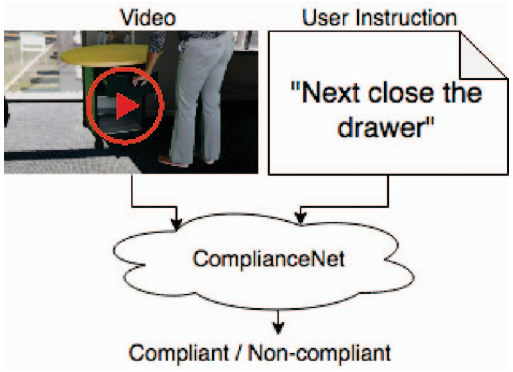


Figure 1: We introduce a novel video-text compliance problem: a framework to learn if the given video is in compliance with a given text instruction.

Here we focus on a different question: how to tell if activities portrayed in a given video match (or *comply with*) a set of natural language instructions. From children building Lego sets, adults assembling IKEA furniture, and technicians repairing machinery [1, 2, 3], to NASA astronauts operating the International Space Station [4], *compliance* with instructions is a quintessential human task. Traditionally, the task of compliance verification is carried out either by human supervisors, or by checking after-the-fact outcomes, which are usually time consuming and costly. In this paper, we introduce the "video-text compliance" problem, a task that verifies if the human activity in a given video is in compliance with a given text instruction (Figure 1).

After careful survey, we recognize that many existing video-related datasets, such as UCF101 [41], Kinetics [10], Charades [39], Something-Something [20], and Moments-

*Equal contribution
†Work done at IBM
‡Internship at IBM

In-Time [30], are either inadequate to be used for the video-text compliance problem, or do not have sufficient safeguards on data privacy. To address this issue, we created the Video-Text Compliance (VTC) dataset. VTC contains 7920 samples, each consisting of a video-text instruction pair and a compliance/non-compliance label. The dataset has over 1.2 million frames. We take a unique approach in data collection so that the dataset can be automatically augmented from a set of core videos. To answer growing concerns on data privacy, we carefully followed privacy preserving safeguards in the generation of VTC dataset. We plan to release VTC dataset to the community to instigate future research in video-text compliance.

Furthermore, we propose ComplianceNet, a novel approach to learn the "video-text compliance" problem. A key insight of our approach is that for this particular compliance problem, we can cast the learning of text instructions as a relation reasoning problem [37]. Hence, instead of a recurrent neural network [18, 13], we can use a multi-scale relation network to learn the *action-object* relations buried in the text instructions. By explicitly aligning the video and text feature spaces, this end-to-end trainable approach achieved 27.5% accuracy performance improvement over the baseline method.

To summarize, the main contributions of this paper include:

- A concrete definition of the video-text compliance problem
- VTC dataset: a new dataset containing triplets of video, text instruction and compliance label
- ComplianceNet: a novel end-to-end trainable deep neural network for compliance verification

The rest of this paper is organized as follows: we review related work in Section 2, followed by the task definition in Section 3. We introduce the VTC dataset in Section 4. In Section 5 we present ComplianceNet, a novel end-to-end trainable compliance network, followed by results in Section 6, before concluding the paper in Section 7.

2. Related Work

Human-Object Interactions (HOI). These methods [11, 19] detect the human, the object and the relationship between the two in a given image. Though this task is similar in flavor to video-text compliance in that it focuses on the person, action and object in the input, HOI does not handle the complexity of text instructions and videos, or verifying compliance.

Visual Question Answering (VQA). VQA is a challenging problem which has received significant attention from the natural language processing and computer vision communities. There are a plethora of proposed methods to infer a correct answer for given questions based on the images [46]. Methods such as [12, 7, 28] utilize attention and RNNs to learn the embeddings from images and text sentences in a common feature space. Most VQA methods involve images, which do not have rich temporal information as videos. They also do not directly address the compliance verification problem.

Action Recognition. Action recognition in videos is a core computer vision problem. Many approaches have been proposed, including two-stream approach [40], inflated 3D convolution (I3D) [10], and approaches combining CNN with LSTM to integrate CNN features over time [15]. More recently, the temporal relation network [47] uses multi-scale relation networks and stochastic sampling across various frames to learn the temporal relations among different frames. In [44] appearance and relation networks are combined to learn actions. Although this area of research is closely related to video-text compliance, it learns only on videos and does not verify compliance with any other input. Video-text compliance learns with natural language instructions, and the instructions could even be unseen at test time, whereas action recognition methods assume a pre-defined set of labels that are seen during training.

Action recognition datasets are the closest usable datasets for video compliance. Numerous datasets are publicly available, such as UCF101 [41], ActivityNet [17], Sports1M [25], Kinetics [10], Charades [39], Moments-In-Time [30] and Something-Something [20]. Many of them are annotated snippets from YouTube and Flickr videos. With increasing public concerns and regulations on data privacy [43], their future usability can be severely constrained. In addition, none of these datasets utilize the automatic augmentation techniques we deploy.

Temporal Alignment of Text with Video. Given a video and a set of written or oral instructions, the task is to map the instructions to relevant sections of the video [9, 29, 16, 23, 6, 31]. Some of these methods do not learn on text, but use them as features. These approaches also assume that the tasks in the videos are carried out correctly. Hence, they differ from the video-text compliance problem in the sense that these approaches do not resolve whether or not the video is compliant with the text instructions.

Application Specific Video Compliance. In [22], the authors address hand washing compliance in a hospital setting. Vision is used to distinguish specific people from the crowd and classify action as disinfectant hand rubbing or not. In [42], the authors propose a computer vision pipeline to recognize products on shelves and verify compliance to the planned layout. However, these papers focus on a single, specific compliance task. Also, neither approach uses a natural language description for verification.

3. Task Definition

Compliance can be defined as the act of obeying a given command. The goal of the "video-text compliance" task is to predict whether the activity shown in an input video complies with an input text instruction. Formally, the input data has two modalities: video $\boldsymbol{V}$ and text instruction $\boldsymbol{T}$. The output is the compliance label defined on the Boolean set:

$$\boldsymbol{V} \times \boldsymbol{T} \rightarrow \{0, 1\} \tag{1}$$

We discuss some issues that arise when learning at the intersection of vision and text for the compliance problem. These discussions are needed to define a workable "video-text compliance" task.

Ambiguities in Language. The instructions $\boldsymbol{T}$ are in text modality. When we rely on natural language to describe a procedure, inevitably we have to deal with the ambiguity of the language itself. As an example, the following sentence represents the typical ambiguous pronoun reference in English grammar [36]: "Albert tells Bob to place his hat on the table". If this instruction is accompanied by a video showing two males each with a hat in his hand, without further contextual information, it is impossible to reason whether the video is in compliance with the instruction because of the ambiguity of the pronoun "his".

Homonyms or Heteronyms. Some common English verbs that are essential to describe instructions have homonyms or heteronyms. An example is *putting*. It is commonly used to describe an action that causes a change (e.g., *putting on shoes*), but it is also commonly used to describe moving a golf club. Training data that doesn't correctly distinguish these actions could confuse the learning system.

Synonyms. There are multiple verbs to explain the same activity. For instance, "open the sliding-door" and "slide the sliding-door" are essentially the same motion, and both verbs "open" and "slide" are legitimate use of language to describe that activity.

Different Motion Patterns. The same verb can have different motion patterns, depending on the agent or objects involved. For instance, "pushing a button" has a very different motion than "pushing a cart". To learn these different motion patterns based on the same text input is a non-trivial problem.

Temporal Complexity. Many daily activities can be temporally decomposed into multiple atomic actions. For example, the activity of "placing a mug into a microwave" can be broken into the following list of atomic actions: 1) *opening* the microwave door, 2) *lifting* the mug, 3) *placing* the item inside the microwave, and 4) *closing* the microwave door. In this paper, we focus on *atomic* actions, similar to the Moments-in-Time [30] and AVA [21] datasets.

Quantitative Measures. Instructions may have quantitative measures, such as "turn the wrench clockwise $270°$." In many applications, whether or not the action meets the quantitative measures is as crucial as performing the action itself. Detecting quantitative movement of objects in videos is yet another non-trivial problem. Although we recognize that this type of quantitative compliance tasks are of great value, they require carefully collected data with great detail.

Noisy Video Data. Video data can be noisy. It could have motion blur, occlusion of the key action steps and objects, and excessive background clutter.

In summary, the utilization of natural language to describe instructions along with video analysis itself brings unique challenges to the "video-text compliance" task. Because of the conciseness of the instructions, there may not be sufficient contextual information to fully resolve the issues raised above. In this paper, we enforce the following restrictions to define a feasible "video-text compliance" problem: **1)** activities must be atomic, no temporally complex actions are allowed; **2)** the video should contain only one person clearly performing one action on one object, i.e. that action or object cannot be implied; **3)** no quantitative measures of actions are allowed in the instruction; and **4)** the text instruction should clearly specify the action and object.

4. The VTC Dataset

The video-text compliance task requires datasets with synchronized triplets of activity videos, text instructions and compliance labels. One possible approach is to add the required labels to existing video datasets, such as Moments-in-Time [30] or Something-Something [20]. However upon careful evaluation, we discovered that this approach is problematic: 1) a large percentage of the videos in these datasets do not meet our criteria outlined in Section 3, i.e. showing one person performing an atomic action on a single object; 2) most of the existing video datasets are crowd-sourced. Therefore, quality control of these datasets do not necessarily meet our constraints; and finally 3) these video datasets have unclear privacy safeguards, which may prevent their future usage [43]. To address these issues, we created the VTC dataset, which we describe in detail here. An overview of the dataset is illustrated in Figure 2.

4.1. Collecting Source Videos

We take a constructive approach to start the data collection by first developing a vocabulary of actions and objects for the videos. 10 distinct actions that overlap between everyday human activities and tasks that appear in repair manuals [4, 3] were chosen. Then, 15 ubiquitous objects used with the chosen actions were selected. The developed list of actions contains antonyms such as *push/pull*, *open/close*, but does not have any synonyms. The objects are of vary-

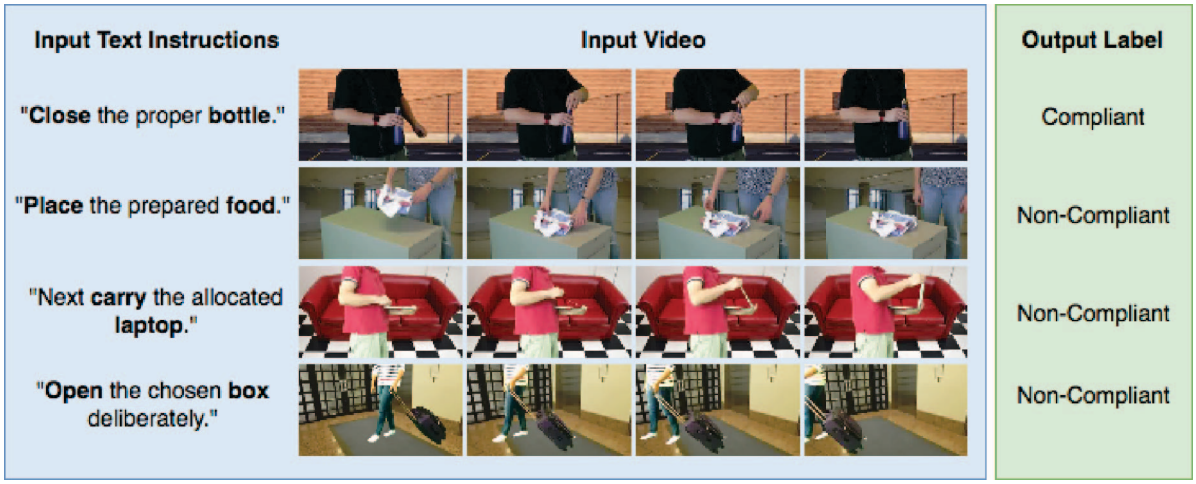
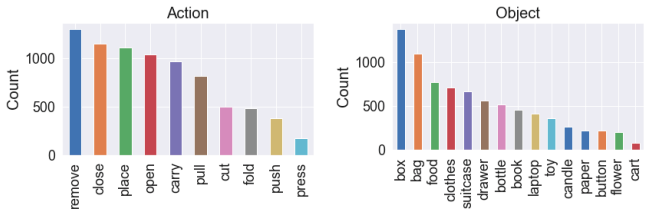


Figure 2: VTC Dataset: Each row in the figure is an example instance from the dataset. Each instance contains a natural language sentence, a video showing human activity, and the corresponding compliance label. Rows $2-4$ are examples representing each type of non-compliance as described in Section 4.3.

ing typical sizes and presentations. Each video contains a person performing only one action, and that action interacts with only one object from the vocabulary. Figure 3 illustrates the histograms of the actions and objects present in our dataset. Although the outer product of the actions and objects, which is the support of video-text compliance task, composes a large space, the real-life occurrences lie in a lower dimensional manifold [20]. We collect videos only from frequently occurring action–object relationships. In addition, to ensure diversity among the action–object pairs, each of the chosen actions appears in combination with at least two or more objects. The same rule also applies to the objects. This requirement has the additional benefit of preventing any trained models from memorizing a particular action–object pair.

A successful method to perform the video-text compliance task should concentrate on the foreground content. Instead of collecting videos in a natural setting, we constructed a filming environment shown in Figure 4a, with professional quality green screens, lighting systems, and a set of video recorders (2 different DSLRs and 2 mobile phone cameras). The filming is carried out by following the protocol: each actor was given an action–object pair such as "open drawer", "carry bag", and instructed to perform the action with the object within $5-10$ seconds. The videos in the VTC dataset are on average 6.5 seconds. Each activity was typically recorded with 2 cameras set up at different heights and angles. In order to preserve privacy, our actors were filmed neck down. As an added precaution, we did not film any identifiable features of the actors such as tattoos, badges, etc. Over a period of time, we collected 792 source videos with green screen on 75 action–object pairs. The action–object tag of each video also serves as the seeds for the semi-automatic generation of text instructions and ground-truth compliance labels, as described in Section 4.4 and Section 4.3 respectively.



(a) (b)

Figure 3: Histogram of (a) actions (b) objects in the VTC dataset.

(a) (b)

Figure 4: (a) An example green screen source video frame. (b) Post-processed video frame.

4.2. Dataset Construction by Auto-Augmentation

The green background of the collected videos provides us great flexibility to scale the dataset by replacing the

green-screen with different background images as post-processing steps (Figure 4b). To achieve this goal, we collected approximately 500 background images. They are in landscape orientation and include natural landscapes, cityscapes, and indoor and outdoor scenes from everyday office and home environments.

To proceed with data augmentation, we first segment the foreground from the source videos using color-based masks. Then each source video is multiplied into 10 videos by super-imposing the foreground activity with 10 distinct background images randomly chosen from the available pool of background images. Furthermore, three independent augmentation techniques were applied to each video: adjustment of brightness, adjustment of contrast, and horizontal flipping. The contrast varies randomly by a ratio within $[-25\%, +25\%]$, while the brightness varies within $[-50, +50]$ (out of a maximum of 255). The horizontal flipping has a probability of 50%.

From the same source videos, we generated two batches of augmented sets, named Batch A and Batch B. The purpose is to provide a utility to test the sensitivity of the developed methods, since both batches have identical foreground content (albeit of different brightness and contrast). Batch A has 69% outdoor background images with the rest being indoor backgrounds, while Batch B has 28% outdoor backgrounds of mostly man-made structures and 72% indoor backgrounds. There is also no overlap between train and test splits within each batch, i.e, the same foreground video does not appear in the training and testing sets. Other than the background images, both batches are identical in terms of text instructions and compliance labeling.

4.3. Labeling

For each of the 7920 videos in each batch, we maintain a record of the ground truth action–object pair. The ground truth action–object pair for each video is the known action and object the actor used when creating the source video. To train and test a model for the compliance task, we need both "compliant" instances of video-text instruction pairs (those where the instruction and video contain the same action–object pair) and "non-compliant" instances of video-text instruction pairs (those where the instruction and video contain a different action-object pairs). In each case, we automatically create an instruction from the action–object pair assigned to the video using the method described in Section 4.4. Next we describe how we assign the action–object pair from which to generate the instruction for each video.

For compliant video-text instruction instances, we use the ground truth action–object pair for the video. For non-compliant video-text instruction instances, we choose an action–object pair that shares with the video's ground-truth action–object pair 1) just the action, 2) just the object or 3) neither the action nor the object. The non-ground truth action or object is selected randomly from a limited set such that all action–object pairs assigned to video instances appear in the overall set of ground truth action–object pairs. For example, the action–object pair selected for a non-compliant instance whose ground truth is "open bottle" might be "open box" (which appears in the set of ground-truth labels for VTC videos), but would not be "open flower" (which does not appear in ground-truth labels for any VTC videos).

Each source video with green screen background is used to generate 10 video-text instruction pair instances. Each instance has a different background. 5 become compliant video-instruction instances, while 5 become non-compliant instances. The non-compliant instances are generated using the following protocol: 1) two instances of random objects with the ground-truth action; 2) two instances of random actions with the ground-truth object, and 3) one instance with a random action–object pair chosen from the set of VTC ground-truth action–object pairs. Note that all 10 instances created from any source video appear either in the test set or training set with no crossover between sets.

4.4. Generating Text Instructions

To enrich the text instructions, we developed a framework to automatically generate natural language instructions from the action–object pairs. Each instruction is an imperative sentence randomly generated from the following template with 7 fields: **head**, **pre-modifier**, **action**, **article**, **adjective**, **object** and **post-modifier**.

The inclusion of the fields other than action and object follows a pre-determined probabilty for that field where the probability is less than 1. The exact word use for that field is randomly chosen from a pre-determined pool. Examples from the pool of **head** include "please", "next", and "then". The **pre-modifier** and **post-modifier** are chosen from the same pool and have a mutual-exclusive probability (i.e., an instruction cannot have both pre- and post-modifier). Each instruction has a maximum of 7 words. As an example, suppose a video is tagged with action–object pair *open–box*, the automatic generator may generate an instruction as: "open the chosen box deliberately". In this case, only the fields of **action**, **article**, **adjective**, **object** and **post-modifier** were chosen by the generator.

5. The ComplianceNet

We designed *ComplianceNet*, a novel end-to-end trainable deep neural network to solve the multi-modal video-text compliance problem. Given video and text instructions as input data, the key design considerations for the network structure are: *representation*, *alignment* and *fusion*. We address each consideration in subsequent sections.

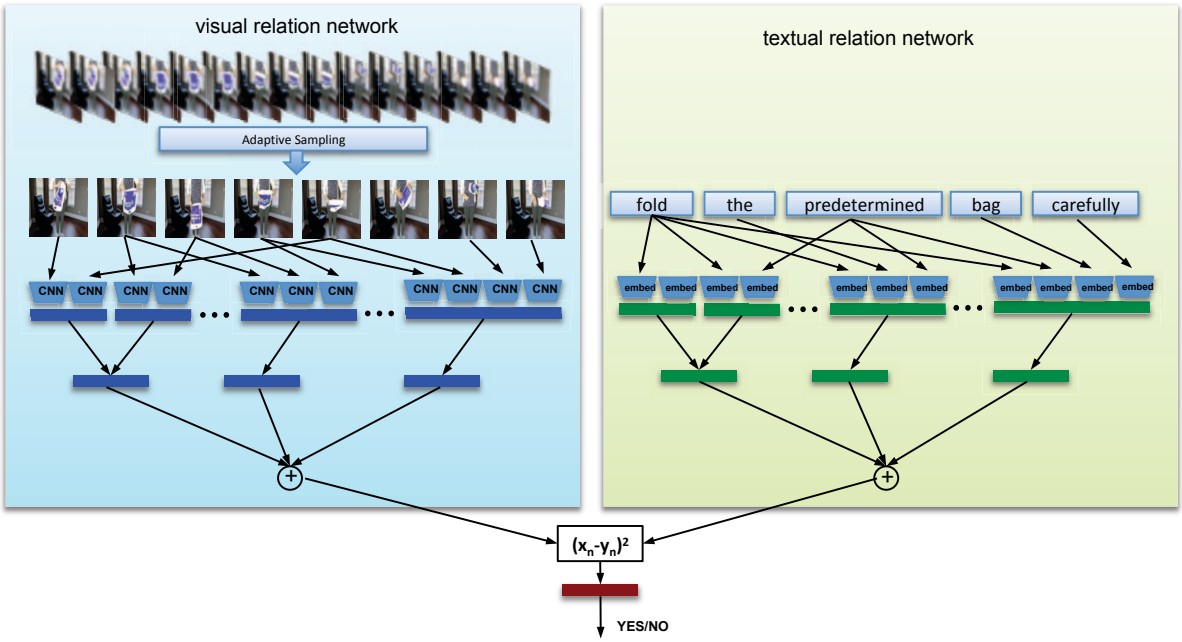


Figure 5: Structure of the ComplianceNet. Two relation networks [47] are used to extract text and video features. These features are aligned and fused to output the probability of compliance. Note that only $2--$, $3--$ and $4--$way relations are shown due to space limitation.

5.1. Representation of the Visual Branch

The visual branch of the ComplianceNet translates a multi-frame video into a feature space. We extract relation features from multiple input video frames using a temporal relation network (TRN) [47]. The details of TRN can be found in [47]. Here we give a brief description, since, it's also related to the text branch of ComplianceNet.

The key concept in TRN is relation reasoning [37]. The relation between the representations of two frames, f_i and f_j is defined as:

$$TR_2(V) = h_\phi(\sum_{i<j} g_\theta(f_i, f_j)) \quad (2)$$

where both $h_\phi(\cdot)$ and $g_\theta(\cdot)$ are trainable multi-layer perceptrons (MLPs). V is the input video with N frames such that $V = \{f_1, f_2, ..., f_n\}$.

TRN also introduced *multi-scale relations*, in which multiple relations among N frames are considered as given in Equation 3.

$$MT_N(V) = TR_2(V) + TR_3(V) + \cdot + TR_N(V) \quad (3)$$

In TRN, only eight frames are used. Furthermore, only up to 3 pairs of each relation are considered in training, which are randomly chosen among all possible k-way combinations. Another level of stochasticity in TRN is on the choice of 8 frames. Instead of selecting them in a deterministic method during training, these 8 frames are randomly selected from all available frames. We hypothesize that these two levels of stochasticity serve as regularization, thus prevent the model from over-fitting. The 8 frames chosen for TRN should sufficiently sample critical frames in a given video. For short videos, such as 3 second videos in Moments-in-Time [30] dataset, randomly sampling 8 frames from 90 frames (3×30 frames per second) may be sufficient. However, the VTC dataset has about 160 frames per video on average. Therefore randomly sampling 8 frames may not capture the entropy in the input video. To address this issue, we propose an adaptive sampling method, outlined in Algorithm 1, to select a subset of the frames. Instead of sampling frames at a given interval, which could lead to oversampling of slow-moving parts of the video, while leaving parts with real actions undersampled, we developed a solution based on the similarity of the adjacent frames [26]. For a given frame j, we construct a cone of similarity of the subsequent frames by comparing the similarity scores of frames j and k. We skip all the intermediate frames as long as the radius of the cone is within a threshold ϵ that is computed automatically for each video.

Algorithm 1 Adaptive Sampling Method

procedure ADAPTIVESAMPLING(N,M) ▷ N: num. of frames in input video, M: desired num. of frames
  for $i = 1$ to $N - 1$ **do**
    Convert frame i and $i + 1$ to gray scale
    Compute similarity score between frame i and $i + 1$ as S_i
  end for
  Compute threshold $\epsilon \leftarrow \frac{1}{M}\left(\max_j \{S_j\} - \min_j \{S_j\}\right)$
  Select frame 1
  $k \leftarrow 1$
  for $i = 2$ to $N - 1$ **do**
    if $\|S_k - S_i\| < \epsilon$ **then**
      Select frame i
      $k \leftarrow i$
    end if
  end for
  return Selected frames
end procedure

5.2. Representation of the Text Branch

The text stream of the ComplianceNet is designed to understand natural language instructions. As discussed in Section 4, the text instructions are imperative sentences anchored by an *action* - *object* pair. Consider the instruction: **close** *the identified* **door** *deliberately*. The *atomic action* (close) and *object* (door) are highlighted. The goal of this branch of ComplianceNet is to learn the presence and ordering of the action-object pair. For an arbitrarily long sentence, it may be necessary to deploy an RNN [18, 13] to learn the representation. However for imperative sentences of fixed maximum length, we show that a relation network can be used to learn the feature representation of the sentence.

Specifically, we tokenize each sentence using GloVe [34], which was pretrained on the corpus of wikipedia2014 and Gigaword 5 [32], with a vocabulary of $400k$ words, 6 $billion$ tokens, and embedding dimension of 300. The details of the text relation network are depicted on the right side of Figure 5.

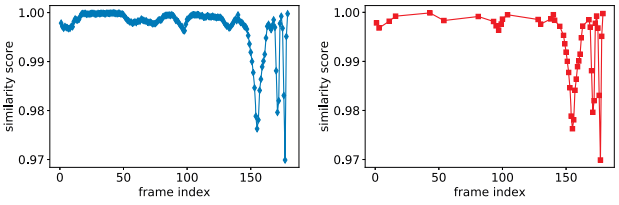
5.3. Alignment and Fusion of Features

The features representations extracted from the visual, $\boldsymbol{r}_v$, and text, $\boldsymbol{r}_t$, branches are 200 dimensional vectors. An explicit alignment layer is used to force the network to learn video and text features in a common feature space. A square difference computation, $\boldsymbol{z} = (\boldsymbol{r}_v - \boldsymbol{r}_t)^2$, is used to minimize the difference between the video and text features, thereby aligning them in feature space. This is similar to the alignment proposed by [45], but encodes relations among multiple frames of video rather than images. The features are finally fused using a fully connected layer with 128 neurons. Finally, the ComplianceNet outputs the probability of compliance, which could be thresholded to obtain a binary outcome. The complete structure of our end-to-end trainable ComplianceNet is shown in Figure 5.

6. Results and Discussion

We implemented ComplianceNet in PyTorch [33] using a publicly available implementation of TRN [47]. All training and inference experiments were conducted on a 36-core x86 server with Xeon E5-2697 CPUs and 4 NVIDIA V100 GPUs.

We first demonstrate the effectiveness of our adaptive sampling method outlined in Algorithm 1. Figure 6a plots the similarity scores of the 180 frames in a video. Figure 6b shows the indices of the 47 selected frames and their corresponding similarity scores. Notice that only a few frames are selected in the flat portion (the first 120 frames) of the video, while almost all later frames are selected. Observe that the envelopes of the two plots are similar, indicating that the adaptive sampling method captures all significant frames in the video. For the VTC dataset, the average number of selected frames per video is 79.8.



(a) (b)

Figure 6: Similarity scores of the (a) original and (b) selected video frames. Similarity scores of the selected frames follow the same trend as the original set of frames.

The visual relation network of the ComplianceNet uses BNInception [24] to extract features from each video frame, as this CNN architecture provides good accuracy and efficiency [47]. We initialize the visual branch of ComplianceNet with weights pre-trained on the Something-Something dataset [20], because similar to VTC, the Something-Something dataset contains many human activity videos. The text branch was trained separately for 20 epochs using the text instructions in the VTC dataset, so that both branches start end-to-end training with non-random weights. The ComplianceNet was trained end-to-end using stochastic gradient descent (SGD) with hyper-parameters as follows: batch size 64, initial learning rate 0.003, momentum 0.9, and weight decay $1. \times 10^{-4}$. After the first 50 epochs, the learning rate was scaled by a factor of 10. We implemented early stopping by terminating the training when the validation loss stopped decreasing for 3 consecutive periods, which is defined as 5 epochs. The validation loss within each period was averaged to smooth out the inherent noise of SGD. We evaluate the performance of ComplianceNet against two methods. The first is a naive Bernoulli process with parameter $p = 0.5$, which is equiva-

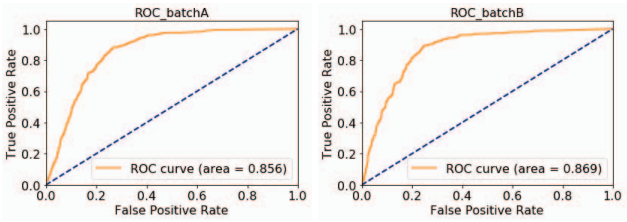
lent to a fair coin toss. The second baseline method predicts the action and object in a given video using TRN and ResNet-152 respectively, and independently predicts the action and object in the text instructions using the Natural Language Toolkit (NLTK) [8]. The baseline model is trained by utilizing the ground truth action and object labels of the VTC dataset. For object classification, we finetune ResNet-152 weights pre-trained on ImageNet. For action recognition, the TRN with BNInception front-end is trained on the VTC dataset starting from random initialization. NLTK was used to parse each instruction. The verb and noun tags in each sentence are interpreted as action and object. A video is inferred as compliant only when the action label from TRN and the object label from ResNet-152 match the output of NLTK.

Method	Accuracy		AUC	
	Batch A	Batch B	Batch A	Batch B
Coin-toss	0.481	0.515	0.488	0.491
Baseline	0.644	0.627	0.644	0.627
ComplianceNet	**0.801**	**0.819**	**0.856**	**0.869**

Table 1: Testing results of models trained on Batch A and tested on Batch A and Batch B. Compared to the baseline trained on TRN + ResNet-152 + NLTK, ComplianceNet improves test accuracy by 27.5% on average.

We trained ComplianceNet with Batch A and tested with Batch A and Batch B of the VTC dataset. Note that the only difference between Batch A and Batch B is that a different set of background images were used for data augmentation. The accuracy and the area under the receiver operating characteristic curve (AUC) on the respective test batches are reported in Table 1. ComplianceNet outputs the probability of compliance. The baseline method outputs 1 when TRN, ResNet-152 and NLTK outputs match, and otherwise outputs 0. Since the output is binary, the baseline method has identical accuracy and AUC values. ComplianceNet improves the baseline results by 27.5% on average between Batches A and B. ComplianceNet's performance on both batches of data are similar even though the batches have different backgrounds. This is an indication that the trained model attends to activity in the foreground of the video, rather than using contextual information in the background. The receiver operating characteristic (ROC) curves of the model for Batch A and Batch B are shown in Figure 7.

We study the sensitivity of the trained model to perturbations in text instructions by changing the action in each instruction to its present continuous form. For example, the instruction "close the proper drawer" becomes "closing the proper drawer". From Table 2, we observe that accuracy degraded by 7.1 percentage points. This accuracy loss could be potentially minimized by using a state-of-the-art word



(a) (b)

Figure 7: ROC curve of test results on (a) Batch A and (b) Batch B.

Dataset	Acc.	AUC
Batch A with perturbed instructions	0.748	0.807
Batch B with perturbed instructions	0.730	0.806

Table 2: Testing results when actions in instructions are converted to present continuous form.

Dataset	Acc.	AUC
Batch A with text reversed	0.587	0.648
Batch B with text reversed	0.576	0.648

Table 3: Test results when the inputs of text instructions are reversed.

embedding [35, 14] that would provide similar language representation for related words.

Since all BNInception layers of the visual relation network are frozen during training, the weights of the convolutional layers are same as TRN weights. As result, we do not visualize the visual relation network using GradCam [38]. Instead, we test the efficacy of ComplianceNet by reversing the order of the text instructions. That is, the sentence "please open the door" becomes "door the open please". The results of this experiment are given in Table 3. We observe significant accuracy degradation, to slightly above the accuracy of coin toss. This illustrates that relation networks learn ordering in relations. So, when the ordering is broken, as in this experiment, the performance deteriorates.

7. Conclusion

In this paper we propose a new type of multi-modal learning problem: "video-text compliance". We presented a constrained video-text compliance problem and constructed the VTC dataset. We also proposed a novel end-to-end trainable network, ComplianceNet, which achieved 27.5% accuracy improvement over the baseline method.

References

[1] Oracle server x7-2 service manual. https://docs.oracle.com/cd/E72435_01/html/E72445/index.html. [Online; accessible 20-March-2019].

[2] Tb8100 base station service manual. http://manuals.repeater-builder.com/2007/TB8000/TB8100_Service_Manual/. [Online; accessible 20-March-2019].

[3] Technician's repair and service manual. http://www.specialtycartz.com/pdf/ezgo_gas.pdf. [Online; accessible 20-March-2019].

[4] International Space Station Operations Checklist, ISS-2A.2A & 2A.2B. http://spaceref.com/iss/operations.html, 2000. [Online; accessible 01-March-2019].

[5] List of best-selling mobile phones. https://en.wikipedia.org/wiki/List_of_best-selling_mobile_phones, 2019. [Online; accessible 22-March-2019].

[6] Jean-Baptiste Alayrac, Piotr Bojanowski, Nishant Agrawal, Josef Sivic, Ivan Laptev, and Simon Lacoste-Julien. Unsupervised learning from narrated instruction videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4575–4583, 2016.

[7] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *The IEEE International Conference on Computer Vision (ICCV)*, December 2015.

[8] Steven Bird, Ewan Klein, and Edward Loper. *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.", 2009.

[9] Piotr Bojanowski, Rémi Lajugie, Francis Bach, Ivan Laptev, Jean Ponce, Cordelia Schmid, and Josef Sivic. Weakly supervised action labeling in videos under ordering constraints. In *The IEEE European Conference on Computer Vision (ECCV)*, 2014.

[10] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017.

[11] Yu-Wei Chao, Yunfan Liu, Xieyang Liu, Huayi Zeng, and Jia Deng. Learning to detect human-object interactions. *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 381–389, 2018.

[12] Kan Chen, Jiang Wang, Liang-Chieh Chen, Haoyuan Gao, Wei Xu, and Ram Nevatia. ABC-CNN: an attention based convolutional neural network for visual question answering. *CoRR*, abs/1511.05960, 2015.

[13] Kyunghyun Cho, Bart van Merrienboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder–decoder approaches. In *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*, pages 103–111, 2014.

[14] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[15] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2625–2634, 2015.

[16] Victor Escorcia, Fabian Caba Heilbron, Juan Carlos Niebles, and Bernard Ghanem. DAPs: Deep action proposals for action understanding. In *European Conference on Computer Vision*, pages 768–784. Springer, 2016.

[17] Bernard Ghanem Fabian Caba Heilbron, Victor Escorcia and Juan Carlos Niebles. ActivityNet: A large-scale video benchmark for human activity understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 961–970, 2015.

[18] Felix A Gers, Jürgen Schmidhuber, and Fred Cummins. Learning to forget: Continual prediction with lstm. 1999.

[19] Georgia Gkioxari, Ross Girshick, Piotr Dollár, and Kaiming He. Detecting and recognizing human-object intaractions. *CVPR*, 2018.

[20] Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fruend, Peter Yianilos, Moritz Mueller-Freitag, et al. The" something something" video database for learning and evaluating visual common sense. In *The IEEE International Conference on Computer Vision (ICCV)*, volume 1, page 3, 2017.

[21] Chunhui Gu, Chen Sun, David A Ross, Carl Vondrick, Caroline Pantofaru, Yeqing Li, Sudheendra Vijayanarasimhan, George Toderici, Susanna Ricco, Rahul Sukthankar, et al. Ava: A video dataset of spatio-temporally localized atomic visual actions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6047–6056, 2018.

[22] Albert Haque, Michelle Guo, Alexandre Alahi, Serena Yeung, Zelun Luo, Alisha Rege, Jeffrey Jopling, Lance Downing, William Beninati, Amit Singh, Terry Platchek, Arnold Milsten, and Fei-Fei Li. Towards vision-based smart hospitals: A system for tracking and monitoring hand hygiene compliance. In *Machine Learning for Healthcare Conference*, pages 75–87, 2017.

[23] Lisa Anne Hendricks, Oliver Wang, Eli Shechtman, Josef Sivic, Trevor Darrell, and Bryan C. Russell. Localizing moments in video with natural language. *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 5804–5813, 2017.

[24] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

[25] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. Large-scale video classification with convolutional neural networks. In *2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1725–1732. IEEE, 2014.

[26] John P Lewis. Fast template matching. In *Vision interface*, volume 95, pages 15–19, 1995.

[27] Yehao Li, Ting Yao, Yingwei Pan, and Hongyang and Chao. Jointly localizing and describing events for dense video captioning. pages 7492–7500, April 2018.

[28] Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. Hierarchical question-image co-attention for visual question answering. In *Advances In Neural Information Processing Systems*, pages 289–297, 2016.

[29] Jonathan Malmaud, Jonathan Huang, Vivek Rathod, Nicholas Johnston, Andrew Rabinovich, and Kevin Murphy. What's cookin'? interpreting cooking videos using text, speech and vision. In *HLT-NAACL*, 2015.

[30] Mathew Monfort, Bolei Zhou, Sarah Adel Bargal, Alex Andonian, Tom Yan, Kandan Ramakrishnan, Lisa Brown, Quanfu Fan, Dan Gutfruend, Carl Vondrick, et al. Moments in time dataset: one million videos for event understanding. *arXiv preprint arXiv:1801.03150*, 2018.

[31] Iftekhar Naim, Young Chol Song, Qiguang Liu, Henry A Kautz, Jiebo Luo, and Daniel Gildea. Unsupervised alignment of natural language instructions with video segments. In *AAAI*, pages 1558–1564, 2014.

[32] Robert Parker et al. *English Gigaword Fifth Edition LDC2011T07*. Linguistic Data Consortium, 2011.

[33] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. In *NIPS-W*, 2017.

[34] Jeffrey Pennington, Richard Socher, and Christopher Manning. GloVe: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

[35] Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*, 2018.

[36] Ann Raimes. *How English works: A grammar handbook with readings*. Cambridge University Press, 1998.

[37] Adam Santoro, David Raposo, David G Barrett, Mateusz Malinowski, Razvan Pascanu, Peter Battaglia, and Timothy Lillicrap. A simple neural network module for relational reasoning. In *Advances in neural information processing systems*, pages 4967–4976, 2017.

[38] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 618–626, 2017.

[39] Gunnar A Sigurdsson, Gül Varol, Xiaolong Wang, Ali Farhadi, Ivan Laptev, and Abhinav Gupta. Hollywood in homes: Crowdsourcing data collection for activity understanding. In *European Conference on Computer Vision*, pages 510–526. Springer, 2016.

[40] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In *Advances in neural information processing systems*, pages 568–576, 2014.

[41] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. UCF101: A dataset of 101 human actions classes from videos in the wild. *CoRR*, abs/1212.0402, 2012.

[42] Alessio Tonioni and Luigi Di Stefano. Product recognition in store shelves as a sub-graph isomorphism problem. In *Proceedings of the International Conference on Image Analysis and Processing*, pages 682–693. Springer, 2017.

[43] Paul Voigt and Axel Von dem Bussche. *The EU General Data Protection Regulation (GDPR)*, volume 18. Springer, 2017.

[44] Limin Wang, Wei Li, Wen Li, and Luc Van Gool. Appearance-and-relation networks for video classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1430–1439, 2018.

[45] Liwei Wang, Yin Li, and Svetlana Lazebnik. Learning deep structure-preserving image-text embeddings. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5005–5013, 2016.

[46] Qi Wu, Damien Teney, Peng Wang, Chunhua Shen, Anthony R. Dick, and Anton van den Hengel. Visual question answering: A survey of methods and datasets. *CoRR*, abs/1607.05910, 2016.

[47] Bolei Zhou, Alex Andonian, Aude Oliva, and Antonio Torralba. Temporal relational reasoning in videos. *European Conference on Computer Vision*, 2018.