Video2Vec: Learning Semantic Spatial-Temporal Embeddings for Video Representation

Sheng-Hung Hu †
School of Computer Engineering
Arizona State University
Tempe, Arizona 85281
Email:shenghun@asu.edu

Yikang Li †
School of Electrical Engineering
Arizona State University
Tempe, Arizona 85281
Email:yikangli@asu.edu

Baoxin Li
School of Computer Science
Arizona State University
Tempe, Arizona 85281
Email:Baoxin.Li@asu.edu

***Abstract*—In this paper, we propose to learn the semantic and temporal embeddings of video clips to represent videos in a high-level way by leveraging the characteristic that videos can be recognized as a temporal sequence of semantically coherent frames. Our architecture extracts CNN features and train two independent Gated Recurrent Unit (GRU) encoders to learn the contextual information of video frames and map both RGB sequence and optical flow sequence into fixed-length representation. Then a Multilayer Perceptron is trained to embed the representation with *word2vec* semantic embedding word vectors. We demonstrate that the proposed architecture can well represent both temporal and semantic information of videos by conducting visual tasks including action recognition, zero-shot classification and "word-to-video" retrieval on the UCF-101 datasets.**

I. Introduction

Video analysis has caught more and more attentions in computer visual community in recent decades but the core problem remains the same – what information does human retrieve from watching videos and how to represent this kind of information?

In this paper, we characterize video as a temporal sequence of semantically coherent visual contents. Thus, we focus on capturing the temporal structures of video clips and embed semantic meanings to form fixed-length vector representation. The purposed representation method aims to bridge the gap between human-level semantic knowledge and the deep-learned visual features.

The deep-learned visual features are extracted from the last fully connected layer from Convolution Neural Network (CNN). CNN has demonstrated power in performing static image recognition tasks [1]. The filter banks in each layer of CNN are capable of acquiring local features in different levels and fuse them with activation function at the end of each layer. The providing features are proved to contain higher-level knowledge compared with local feature descriptors, like HOG3D and Dense Trajectory [2], [3].

To deal with the temporal structure among features, one naive approach is to take each video frame as an independent static image and average at the video level to obtain the corresponding vector representation. However, this approach completely ignores the temporal frame order and treats each frame equally important, thus the produced representation would be incomplete and unable to tell the whole story.

In order to learn the temporal structure of videos, we utilize the Recurrent Neural Networks (RNNs). RNNs are capable of preserving relations of each atom in learning sequential information since their specific structure of layers, which the current neuron outputs depend on the previous computations. However, RNNs are also known to suffer from the vanishing gradient problem especially when the structure goes deeper. Therefore, two well-known gated structures are provided as better options – the Long-Short Term Memory (LSTM) and Gated Recurrent Unit (GRU) [4], [5]. These two structures avoid gradient vanishing by implementing "gate units" which decide what to "forget" and what to "memorize" depended on the different gate values. In our architecture, we choose GRUs to encode video sequences because they have comparative performance with LSTM while maintaining fewer gates and thus less computation cost.

The learned vector representation, although temporally encoded, still fails to gap human-level semantic knowledge and deep-learned visual features due to the fact that the representation lacks of human-level descriptions. Therefore, we embed semantic meanings to the encoded representation by learning the projection between the representation and *word2vec* Semantic Embedded Space (SES) [6]. The SES is learned from Wikipedia documents with more than 1 billion words and have proven its ability to contain semantic meanings of textual words [6]. This embedding enables us to utilize the power of words and allow our encoded representation to "semantically associate".

We evaluate our video representation on three tasks on the well-known UCF-101 dataset [7]. To demonstrate our temporal and semantic embedding video representation can successfully capture the spatial-temporal information, we perform 1). the action recognition task. To show the representation is semantically embedded, we perform 2). Zero-shot classification and 3). "semantic video retrieval".

The rest of paper is organized as follow: In section II, we review state-of-the-art works on action recognition, zero-shot learning and video retrieval. In section III, we introduce the architecture of our semantic spatial-temporal embedding system. Results and details of the three tasks are shown in

† These two authors have equal contributions.

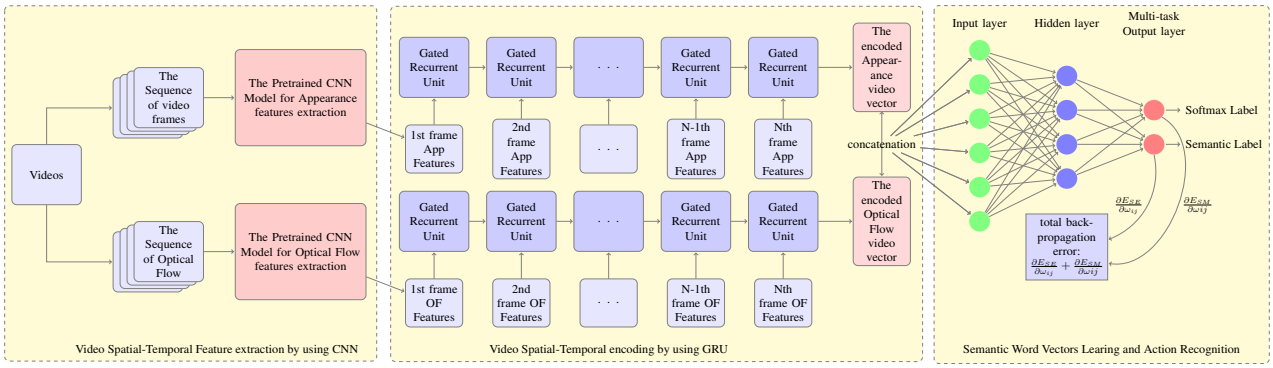

Fig. 1: Framework of our temporal and semantic embedding video representation architecture.

section IV. Finally we draw a conclusion in section V.

II. Related Work

Conventional action recognition tasks utilize *hand-crafted* descriptors such as dense features to capture spatial-temporal information and use high dimension vector encoding method like fisher vector encoding to represent video [2], [3]. This approach can achieve effective representations and is robust to background clutter, illuminance changes and video noise as well. However, hand-crafted dense local features lack semantic and discriminative capacity and cannot represent high-level informations.

As a result, using high-level CNN features or combine multiple level features to represent videos has become popular in recent years. Simonyan *et al* [8] introduce a method to fuse both CNN appearance and optical flow features to form representation while Wang *et al* [9] propose descriptors that pool both low-level hand-crafted features and high-level deep-learned features to represent videos. Their representations achieve great success in performing classification and recognition tasks as they benefit from the use of both high and low level features.

One limitation of representing videos by fusing multiple level features is that they fail to fully leverage temporal information of the given video clips. To overcome this limitation, several approaches [10]–[12] suggest to encode video temporal information by Long Short Term Memory (LSTM). LSTM can be considered as a gated Recurrent Neural Network (RNN) which is capable of discovering the implicit structure of input sequences in the time domain while avoid gradient vanishing effect: ability to backpropagate error signals decrease when dealing with long-range temporal sequences. Representing videos by LSTM show advantages when modelling complex temporal dynamics and demonstrate competitive recognition results compared with the state-of-the-art multiple level features method.

Unfortunately, none of the proposed LSTM encoded video representations consider embedding human-level semantic meanings to their representation – thus do not allow their representation to "semantically associate" and conduct high-level semantic task such as semantic video retrieval: retrieve video contents by unknown word descriptions, and zero-shot classification: recognize unseen video categories.

The concept of semantic video retrieval has been proposed for decades, most widely used approaches take advantage on the high-level or mid-level representation of video such as content-based video retrieval or latent Dirichlet allocation representations [13]–[16]. However, no previous work using deep neural network embedded with the semantic meaning of video labels or descriptors. Although [11], [17], [18] can generate sequences based on the video contents, the semantic meaning of video is not learned from the video linguistic descriptors itself but from an additional dictionary. Therefore, we proposed to embed linguistic information directly to the temporally encoded video representation at by learning the mapping function between representation and label vectors in *word2vec* semantic space.

Most existing zero-shot learning techniques focus only on static images and take advantages of attributed-based semantic space: classify unseen categories by trained attribute model [19]. However, it is rather hard to obtain sufficient and reliable attribute-based representation for large-scale dataset. Our representations provide more direct connection between human-level semantic meanings and temporally encoded representation and outperform state-of-the-art method [19].

III. Method

Given a massive collection of video clips with semantic labels, our goal aims at capturing the temporal structure of given video clips and embed the semantic meanings of labels to obtain fixed-length vector representation. The proposed framework is demonstrated in Fig. 1. Our approach consists of three major parts: 1. extract spatial and temporal features from CNNs, 2. encode videos' temporal structures by GRUs and 3. learn the projection function between encoded representations and their corresponding vectors in the semantic embedding space by MLP. Our proposed architecture is an end-to-end learning model which can be trained by backpropagation with loss computed by summing the result of Hinge rank loss function and softmax loss function at the output layer of our model.

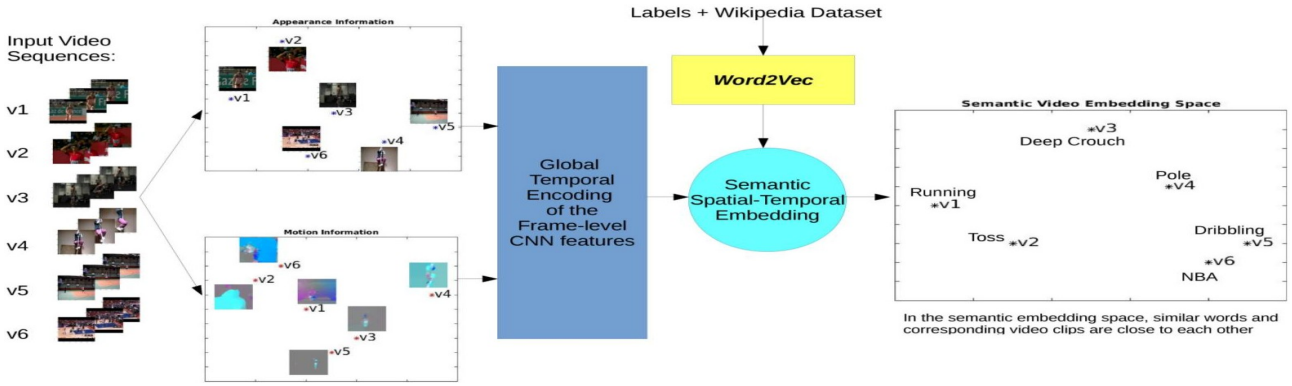

Fig. 2: Concept of our model on embedding semantic meaning to temporally encoded representation. Videos clips carry similar semantic meanings can vary greatly when temporally encoded (e.g., "Basketball dunk" and "Basketball" videos are far from each other when only temporal encoding is performed) . By learning the projection matrix to embed human-level semantic knowledge, our purpose representation can better represent videos in high-level. (e.g., "Basketball dunk" and "Basketball" videos are projected to similar coordinate after embedding semantic knowledge).

A. CNNs to extract spatial-motion features sequences

With a collection of video clips V where each clip $v \subset V$ contains a sequence of frames with a specific order $\{f_1, ..., f_n\}$ and label l_v. We first pull out rgb frames to represent spatial information and compute the dense optical flows upon the pulled-out rgb frames to represent motion information. Noticed that both dense optical flows and rgb frames are computed at 10fps and optic flows are computed by using the popular implementation described in [20].

Two CNN pre-trained models are then used to extract appearance f_{v_app} and optical flow features f_{v_of} respectively. The "VGG-f" [21] pre-trained model on the ImageNet ILSVRC-2012 dataset [22] is responsible for extracting appearance features while the pre-trained networks implemented by [23] is used to extract optical flow features. We collect f_{v_app} and f_{v_of} from the last rectified linear unit (ReLU) layer from each pre-trained model.

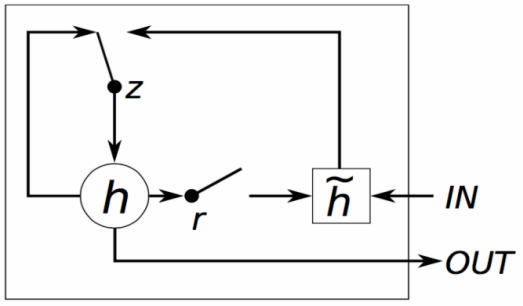

Fig. 3: A standard Gated Recurrent Unit [5]. GRU memorizes a state variable h. The state variable can be either update or remain the same depends on the value of the update gate z. The reset gate r controls the influence of previous input sequence toward current input.

B. GRUs to encode spatial-motion feature sequences

Given the spatial-temporal feature sequences f_{v_app} and f_{v_of}, we encode each of them independently with two variable-length GRUs. The choice of encoding separately is based on the hypothesis that f_{v_app} and f_{v_of} contain different knowledge of the video clips regarding the order of frames and should not be pooled together at frame level.

The Gated Recurrent Unit (GRU) uses reset gate r^n and update gate z^n (both gates take values between zero and one) to memorize and forget sequence state. Thus avoiding the gradient vanishing problem while maintaining the power to discover temporal relationship. Given an input sequence $x = (x_1, ..., x_N)$, the GRU encoder feed forward the input by iterating the following equation from n =1 to N:

$$r^n = \sigma(W_r x_n + U_r h^{n-1}) \quad (1)$$

$$z^n = \sigma(W_z x_n + U_z h^{n-1}) \quad (2)$$

$$\tilde{h}^n = tanh(W x_n + U(r^n \bullet h^{n-1}) \quad (3)$$

$$h^n = (1 - z^n) \bullet h^{n-1} + z^n \bullet \tilde{h}^n \quad (4)$$

where h^n is the model state at step n, $\tilde{h}^n$ is the purposed state update at step n, (σ) denotes the logistic sigmoid function, $(\bullet)$ denotes element-wise product and W_r, W_z, U_R, U_z denote hidden variables [5].

The GRU architecture we use in this paper is shown in the middle part of Fig. 1, where each GRU contains 1024 hidden units. We experienced with various numbers of units and choose 1024 to best trade off complexity and performance. At the end of GRUs outputs, we implement a mean-pooling layer to obtain a fixed-length representation. The output of our GRU encoder are the appearance video representation E_v^{app} and optical flow video representation E_v^{of}. Both have 1024 dimension .

Finally, a weighted concatenation is performed to combine the appearance and optical flow representation at video level:

$E_v = [(\frac{1}{2})E_v^{app}; (\frac{3}{2})E_v^{of}]$. We choose the weight to be $\frac{1}{2}$ and $\frac{3}{2}$ as recommended in the work [11]. The result temporally encoded representation E_v has 2048 dimension.

C. MLP to embed semantic meanings to video representation

In order to embed human-level semantic knowledge to the encoded video representation E_v, a Multilayer Perceptron (MLP) is trained to serve as the mapping function $g(\bullet)$ that project E_v to its corresponding coordinate $g(E_v)$ in the *word2vec* transformed 500 dimension Semantic Embedded Space.

Semantic label vectors $m = (m_1, ..., m_N)$ are obtained by feeding forward the given video labels $l = (l_1, ..., l_N)$ through the *word2vec* network: $m_n = M(l_n)$. The *word2vec* network $M()$ is a skip-gram text model trained on Wikipedia documents that containing 5.4 billion words. For labels that contains multiple words, such as "riding horse" or "rock climbing indoor" we generate a single vector z_n by Bag-of-Words (BoW) method – averaging word vectors for every unique words in the label l_n: $m_n = \frac{1}{w}\sum_{w \subset l_n} M(l_n)$.

Two distinct loss functions are implemented and errors from both loss functions are back-propagated to train the whole model. To embed semantic label vectors m_n, we use the hinge rank loss function given as:

$$loss_1 = -\sum_{j \neq n} max[0, margin - m_n \cdot g(E_v) + m_j \cdot g(E_v)] \quad (5)$$

The second loss function serves as a fine-tuning part and is effective when performing video recognition task. It measures the probability of $g(E_v)$ belonging to a specific label l_v through a softmax layer. The second loss function is defined as:

$$loss_2 = -\sum_{n=1}^{N} \{l_v = n\} log(P(l_v = n | g(E_v))) \quad (6)$$

$$P(l_v = n | g(E_v)) = \frac{e^{g(E_v) \cdot w_v}}{\sum_{n=1}^{N} e^{g(E_v) \cdot w_n}} \quad (7)$$

where $\{l_v = n\}$ equals to 1 when the video clip label l_v belongs to the n category and 0 otherwise. w is the softmax layer weights.

Errors from both loss functions are summed together when performing backpropagation and weights update, hence each loss function can serve as a compensation and fine-tuning method for the other.

Fig. 2 illustrates the concept of our semantic embedding model. Video clips with similar semantic informations can be temporally encoded differently since the GRU encoders lack of human-level semantic knowledge. After embedding semantic knowledge by utilizing *word2vec* algorithm, video clips which share similar semantic meanings will be project to close coordinate in semantic space.

IV. Result

We evaluate the proposed video representation architecture on UCF-101 dataset [7] (13320 video clips from 101 categories) by conducting three visual tasks: 1. video action recognition, 2. video zero-shot learning, and 3. semantic video retrieval. All three tasks share the same architecture setting. Dense optical flow and rgb frames are extracted every 0.1 second (10fps). Each GRU unit contains 1024 hidden units. Size of the MLP input, hidden and output layer is 2048, 1200 and 500 respectively. Learning rate for the whole end-to-end structure is initialized as 0.0001 for the first 15 epochs and then reduced by half for every 15 epochs. Batch size is set to 30 videos per batch. The margin for Hinge rank loss computation, however, varies between tasks: 0.4 for zero-shot classification and 0.55 for video recognition and semantic video retrieval.

TABLE II: 3-fold recognition accuracy on UCF-101 dataset

Algorithm	**Accuracy (%)**
Dense Trajectory (Wang *et al*) [3]	75.1
LRCN (Donahue *et al*) [11]	82.9
Composite LSTM model (Srivastava *et al*) [10]	84.3
Two-stream Convolution Net (Simonyan *et al*) [8]	88.0
Deep LSTM with 30 Frame (Ng *et al*) [12]	88.6
TDDs (Wang *et al*) [9]	**91.5**
Our result (softmax)	87.5

TABLE III: Zero-shot learning accuracy on UCF-101 dataset

Algorithm	**Accuracy (%)**
ConSE (Norouzi *et al*) [24]	10.5
SES (Xu *et al*) [25]	10.9
IAP (Lampert *et al*) [19]	12.8
DAP(Lampert *et al*) [19]	14.3
Our result	**14.7**

A. Video Action Recognition

For video action recognition task, we follow the three train-test split rule in [7] to train and test our video representation architecture. When testing, we categorize test videos to the trained label that has maximum probability based on the softmax layer output.

We compare our result performance with [3] and [8]–[12] as shown in Table II. Compared with [8], [9], our video representation does not require any kind of early or late classifier-based fusion or pooling method. Compared with [12], our video representation does not train multi-layer LSTMs and thus more time efficient. We believe that applying late fusion and multi-layer GRUs can potentially lead to further improvements. However, in this paper, we focus on demonstrating that our video representation can handle much high-level task.

B. Video Zero-Shot Learning

For video zero-shot classification task, we perform a 10-fold cross-validation where, in each fold, we split 101 categories into two sub-datasets: training dataset with 51 categories and testing dataset with the rest 50 categories. When testing,

TABLE I: Query word pool and their retrieval results. Query words are never seen in training dataset. Retrieve results are the retrieved video clips denoted by their categories in UCF-101 dataset. Number in the brackets denoted how many retrieved video clips belong to this category in the top10 hit list.

Query Labels	**Top10 Retrieve Results**	Query Labels	**Top10 Retrieve Results**
NBA	Basketball Dunk (10)	Extreme	Rock Climbing Indoor (5), Uneven Bars (2), Soccer Juggling (2), Pole Vault (1)
Orchestra	Playing Cello (9), Playing Piano (1)	Tide	Cliff Diving (4), Surfing (2), Throw Discus (2), Sky Diving (1), Rafting (1)
Army	Military Parade (10)	India	Paying Tabla (4), Playing Sitar (2), Head Massage (1), Cricket Shot (1), Mixing (1)
Music	Playing Sitar (9), Playing Piano (1)	Celebrate	Military Parade (6), Long Jump (1), Band Marching (1), Ice Dancing (1), Blowing Candles (1)
Computer	Typing (10)	Home-run	Baseball Pitch (5), Basketball Dunk (3), Field Hockey Penalty (1), Frisbee Catch (1)
Park	Biking (9), Golf Swing (1)	Boat	Kayaking (4), Rafting (2), Rowing (2), Cliff Diving (1), Push Ups (1)
Summit	Cliff Diving (7), Skiing (2), Rope Climbing (1)	Toy	Yo-yo (4), Nun chucks (4), Pull Ups (1), Juggling Ball (1)
School	Skate Boarding (10)	Snow	Skiing (2), Ice Dancing (2), Cricket Bowling (1), Pole Vault (1), Blowing Candles (1), Blow Dry Hair (1), Rafting (1), Sky Diving (1)
Park	Biking (9), Golf Swing (1)	Acrobatics	Juggling Balls (5), Soccer Juggling (5)
Water	kayaking (10)	Ocean	Cliff Diving (4), Sky Diving (3), Kayaking (2), Rafting (1)
FIFA	Soccer Penalty (8), Soccer Juggling (2)	Hurl	Throw Discus (2), Mopping Floor (2), Baby Crawling (1), Javelin Throw (1), Cricket Shot (1), Blowing Candles (1), Pull Ups (1)
Club	Golf Swing (8), Soccer Juggling (2)	Hiking	Biking (5), Kayaking (4), Rafting (1)
Nature	Tai Chi (7), Hammering (2), Walking with Dog (1)	Swim	Diving (5), kayaking (3), Cricket Bowling (1), Sky Diving (1)
Beethoven	Playing Cello (8), Playing Voilin (2)	Jogging	Biking (5), Skate Boarding (2), Soccer Juggling (1), Skiing (1), Ice Dancing (1)
Classical	Playing Cello (7), Playing Voilin (3)	Foam	Blowing Candles (7), Pull Ups (1), Rope Climbing (1), Juggling Balls (1)
Yankees	Baseball Pitch (10)	Hip-hop	Trampoline Jumping (6), Swing (4)
Duel	Boxing Punching Bag (8), Punch(2)	Scramble	Pull Ups (6), Trampoline Jumping (2), Rope Climbing (1), Cricket Shot (1)
Lifting	Body Weight Squats (4), Rope Climbing (4), Pull Ups (2)	Mat	Rope Climbing (4), Pommel Horse (3), Trampoline Jumping (2), Javelin Throw (1)
Martial	Fencing (3), Archery (3), Boxing Punching Bag (3), Balance Beam (1)	Parachuting	Diving (6), Cricket Bowling (2), Hand Stand Walking (1), Sky Diving (1)
Tumbling	Trampoline Jumping (8), Throw Discus (1), Frisbee Catch (1)	Hunting	Horse Riding (3), Kayaking (3), Nun chucks (3), Frisbee Catch (1)

we categorize test videos to the "unseen" test label that has maximum cosine similarity (Nearest Neighbor) to our temporal and semantic embedded representation.

We compare our results performance with [19], [24], [25] as shown in Table III. The proposed representation outperforms [19], [24], [25] by around 2% and 4% and slightly outperforms the state-of-the-art attribute-based model DAP [19] by around 0.5%. The superior performance of our proposed representation indicates the effectiveness of embedding human-level semantic meanings to temporally encoded video sequence.

C. Semantic Video Retrieval

To further demonstrate that our purposed representation can perform "semantically associate", we challenge it with the semantic video retrieval task. For this task, we follow the first train-test split rule in [7] to separate the UCF-101 dataset. A word pool which contains 40 words were created manually to serve as query words. When testing, we first feed forward all test videos through previous trained architecture and obtain their temporal and semantic embedding representation. Then, for each query word, our architecture retrieve top 10 test video clips that have maximum cosine similarity (Nearest Neighbor) to the *word2vec* transformed query word as demonstrated in Fig. 4. Noticed that, for query which contains multiple words, we perform the same Bag-of-Words method as described in the method section.

The query word pool and their retrieval results are shown in Table 3. Our model presents good capability in capturing both video contents and their semantic knowledges. One simple

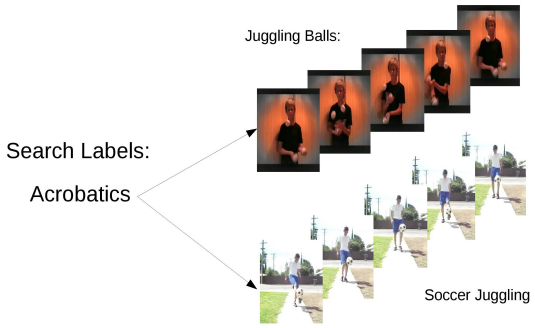

Fig. 4: Example result of our architecture performing the semantic video retrieval task. By inputing the query word "Acrobatics", our architecture retrieves "Juggling Balls" and "Soccer Juggling" video clips.

example is the retrieved results using the query word "NBA". All retrieved videos belongs to the "Basketball Dunk" category of UCF-101 dataset. Another more complex searching word example is "Acrobatics" which gives us "Juggling Balls" and "Soccer Juggling".

V. Conclusion

In this paper, we proposed an effective video representation architecture by learning the semantic spatial-temporal embeddings of video clips. Gated Recurrent Units (GRUs) are utilized to encode CNN features in time domain and MLP is trained to embed human-level semantic knowledge. We evaluate our representation architecture on the UCF-101 dataset for action recognition, zero-shot classification and semantic video retrieval. Our result proves our video representation architecture can achieve comparative performance in action recognition compared with the state-of-the-art methods while outperforms previous works in zero-shot classification. For semantic video retrieval task, our architecture demonstrate its ability to perform "semantically associate" by retrieving videos by "untrained" word query.

Acknowledgment

The work was supported in part by ONR grants N00014-15-1-2344 and N00014-15-1-2722. Any opinions expressed in this material are those of the authors and do not necessarily reflect the views of ONR or ARO.

References

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems 25*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2012, pp. 1097–1105. [Online]. Available: http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf

[2] A. Kläser, M. Marszałek, and C. Schmid, "A spatio-temporal descriptor based on 3d-gradients," in *British Machine Vision Conference*, sep 2008, pp. 995–1004. [Online]. Available: http://lear.inrialpes.fr/pubs/2008/KMS08

[3] H. Wang, A. Kläser, C. Schmid, and C.-L. Liu, "Action Recognition by Dense Trajectories," in *IEEE Conference on Computer Vision & Pattern Recognition*, Colorado Springs, United States, Jun. 2011, pp. 3169–3176. [Online]. Available: http://hal.inria.fr/inria-00583818/en

[4] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[5] K. Cho, B. van Merrienboer, Ç. Gülçehre, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," *CoRR*, vol. abs/1406.1078, 2014. [Online]. Available: http://arxiv.org/abs/1406.1078

[6] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[7] K. Soomro, A. R. Zamir, and M. Shah, "UCF101: A dataset of 101 human actions classes from videos in the wild," *CoRR*, vol. abs/1212.0402, 2012. [Online]. Available: http://arxiv.org/abs/1212.0402

[8] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," *CoRR*, vol. abs/1406.2199, 2014. [Online]. Available: http://arxiv.org/abs/1406.2199

[9] L. Wang, Y. Qiao, and X. Tang, "Action recognition with trajectory-pooled deep-convolutional descriptors," *CoRR*, vol. abs/1505.04868, 2015. [Online]. Available: http://arxiv.org/abs/1505.04868

[10] N. Srivastava, E. Mansimov, and R. Salakhutdinov, "Unsupervised learning of video representations using lstms," *CoRR*, vol. abs/1502.04681, 2015. [Online]. Available: http://arxiv.org/abs/1502.04681

[11] J. Donahue, L. A. Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell, "Long-term recurrent convolutional networks for visual recognition and description," in *CVPR*, 2015.

[12] J. Y. Ng, M. J. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici, "Beyond short snippets: Deep networks for video classification," *CoRR*, vol. abs/1503.08909, 2015. [Online]. Available: http://arxiv.org/abs/1503.08909

[13] H. Li, L. Liu, F. Sun, Y. Bao, and C. Liu, "Multi-level feature representations for video semantic concept detection," *Neurocomputing*, vol. 172, pp. 64–70, 2016.

[14] M. Gitte, H. Bawaskar, S. Sethi, and A. Shinde, "Content based video retrieval system," *Int J Res Eng Technol*, vol. 3, no. 6, 2014.

[15] R. Veltkamp, H. Burkhardt, and H.-P. Kriegel, *State-of-the-art in content-based image and video retrieval*. Springer Science & Business Media, 2013, vol. 22.

[16] C. G. M. Snoek and M. Worring, "Concept-based video retrieval," *Found. Trends Inf. Retr.*, vol. 2, no. 4, pp. 215–322, Apr. 2009. [Online]. Available: http://dx.doi.org/10.1561/1500000014

[17] 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.

[18] 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.

[19] C. Lampert, H. Nickisch, and S. Harmeling, "Attribute-based classification for zero-shot visual object categorization," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 36, no. 3, pp. 453–465, March 2014.

[20] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert, "High accuracy optical flow estimation based on a theory for warping," in *Computer Vision-ECCV 2004*. Springer, 2004, pp. 25–36.

[21] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, "Return of the devil in the details: Delving deep into convolutional nets," in *British Machine Vision Conference*, 2014.

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

[23] G. Gkioxari and J. Malik, "Finding action tubes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 759–768.

[24] M. Norouzi, T. Mikolov, S. Bengio, Y. Singer, J. Shlens, A. Frome, G. S. Corrado, and J. Dean, "Zero-shot learning by convex combination of semantic embeddings," *arXiv preprint arXiv:1312.5650*, 2013.

[25] X. Xu, T. Hospedales, and S. Gong, "Semantic embedding space for zero-shot action recognition," in *Image Processing (ICIP), 2015 IEEE International Conference on*. IEEE, 2015, pp. 63–67.