Paper Seminar Presentation: Two-Stream RNN/CNN for Action Recognition in 3D Videos

R. Zhao, H. Ali, and P. van der Smargt, “Two-Stream RNN/CNN for Action Recognition in 3D Videos,” arXiv preprint arXiv:1703.09783, 2017

Project: Query by Video for Surgical Activities

Presenter: Gianluca Silva Croso

Team member: Felix Yu

Mentors: Tae Soo Kim, Dr. Swaroop Vedula, Dr. Gregory Hager, Dr. Haider Ali

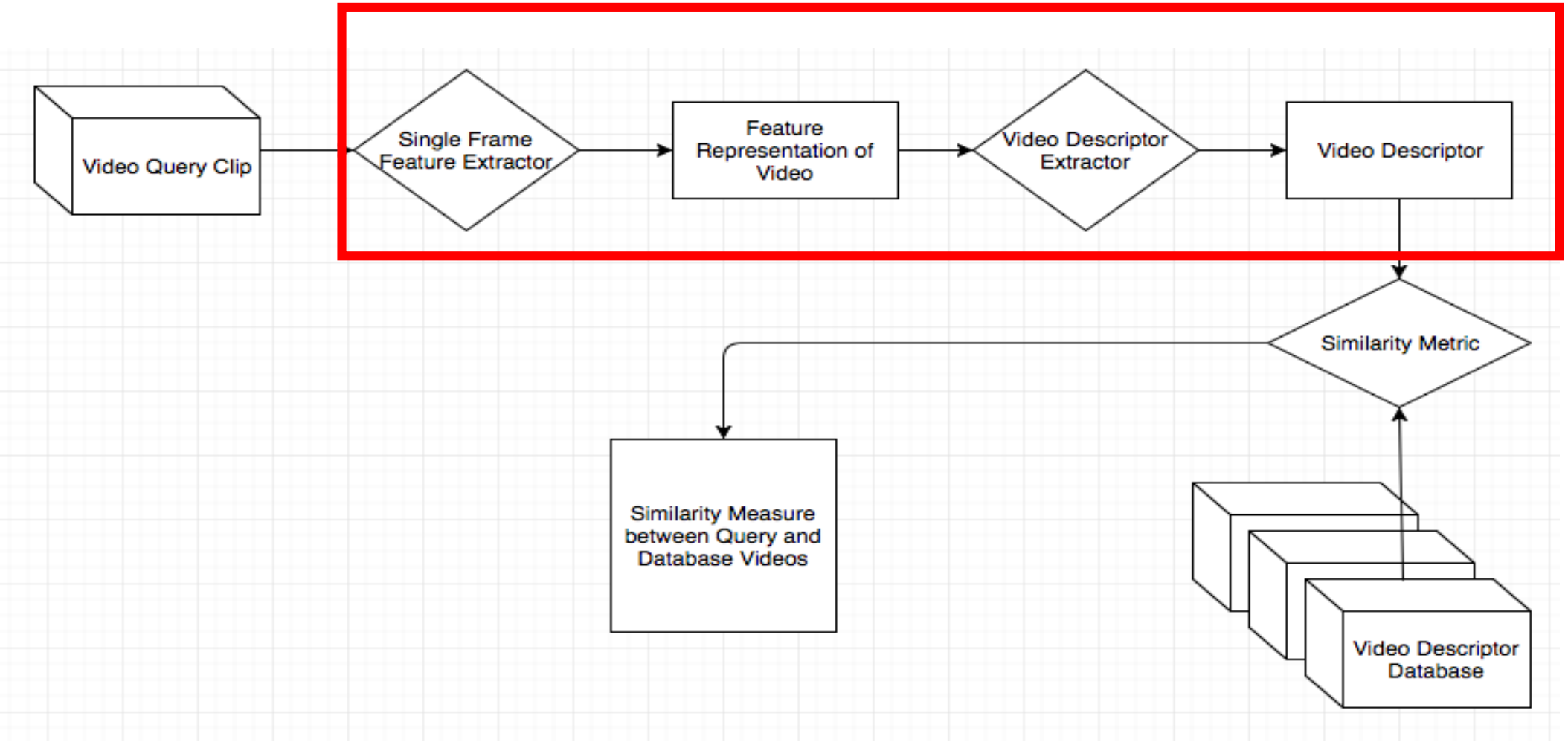
Project Overview

- Design Machine Learning pipeline to query for similar surgical video from a database
- Resulting video should be of the same activity
- Cataract Surgery

Query Video

Closest Matching

Project Overview



Paper Selection

- Zhao, R., Ali, H. and van der Smargt, P. (2017). Two-Stream RNN/CNN for Action Recognition in 3D Videos.

- Our query by video is ultimately a form of activity recognition
- Video based learning needs to capture spatial **and** temporal information, as well as their relationship
  - state-of-the-art network architectures using CNNs and RNNs
  - possible ways of conciliating/fusing both types of information

Summary of Problem and Significance

- Problem: Action Recognition
  - 60 different day-to-day, health related, or mutual actions

- Significance:
  - Action recognition is currently a big research area
  - Applicable to several fields
    - Health monitoring
    - Assisted living
    - Surveillance
    - Robot perception and cognition

Key contributions and Results

- Novel RNN structure
  - faster convergence, lower computational costs
- Two fusion methods for RNN and CNN data
  - Decision fusion and feature fusion
- Improved recognition rate on state-of-the art methods by 14%

Background: RNN

- Recurrent Neural Networks
- Capture relationship between inputs over multiple time-steps

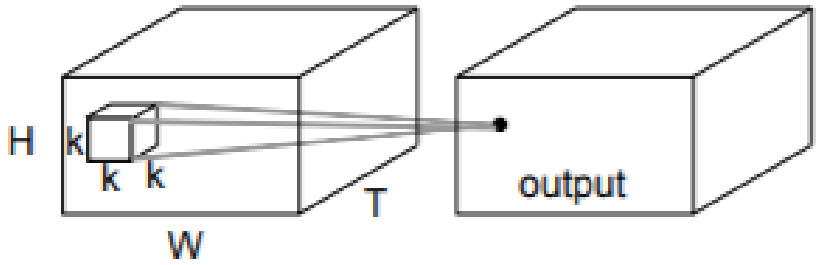
$$\mathbf{h}_t = \sigma\left(\mathbf{W}\begin{pmatrix}\mathbf{x}_t \\ \mathbf{h}_{t-1}\end{pmatrix}\right)$$

$$\mathbf{y}_t = \sigma\left(\mathbf{V}\mathbf{h}_t\right)$$

- Problem: remember information over **long** time (vanishing gradients)
- LSTM: Long Short-Term memory
  - “gated cells” improve information retainment over time
- GRU: Gated Recurrent Unit
  - Simplifies LSTM to be faster and more memory-efficient
- Bidirectional RNN
  - Perform a forward pass and a backward pass over the data

Background: CNN

- Convolutional Neural Networks



- Network architecture based on convolutional filters to learn spatial features on progressively more general scale

- Video: 3D convolutions with temporal dimension allows learning spatiotemporal information across *small* number of timesteps

Background: Other concepts

- Batch Normalization:
  - Statistical technique to speed up convergence, stabilizes model
- Support Vector Machine:
  - Multi-class classifier that attempts to maximize the boundaries between distinct classes
- Fine-Tuning:
  - Training a neural-network that has been initialized with pre-trained weights based on another dataset

Experiment: Data

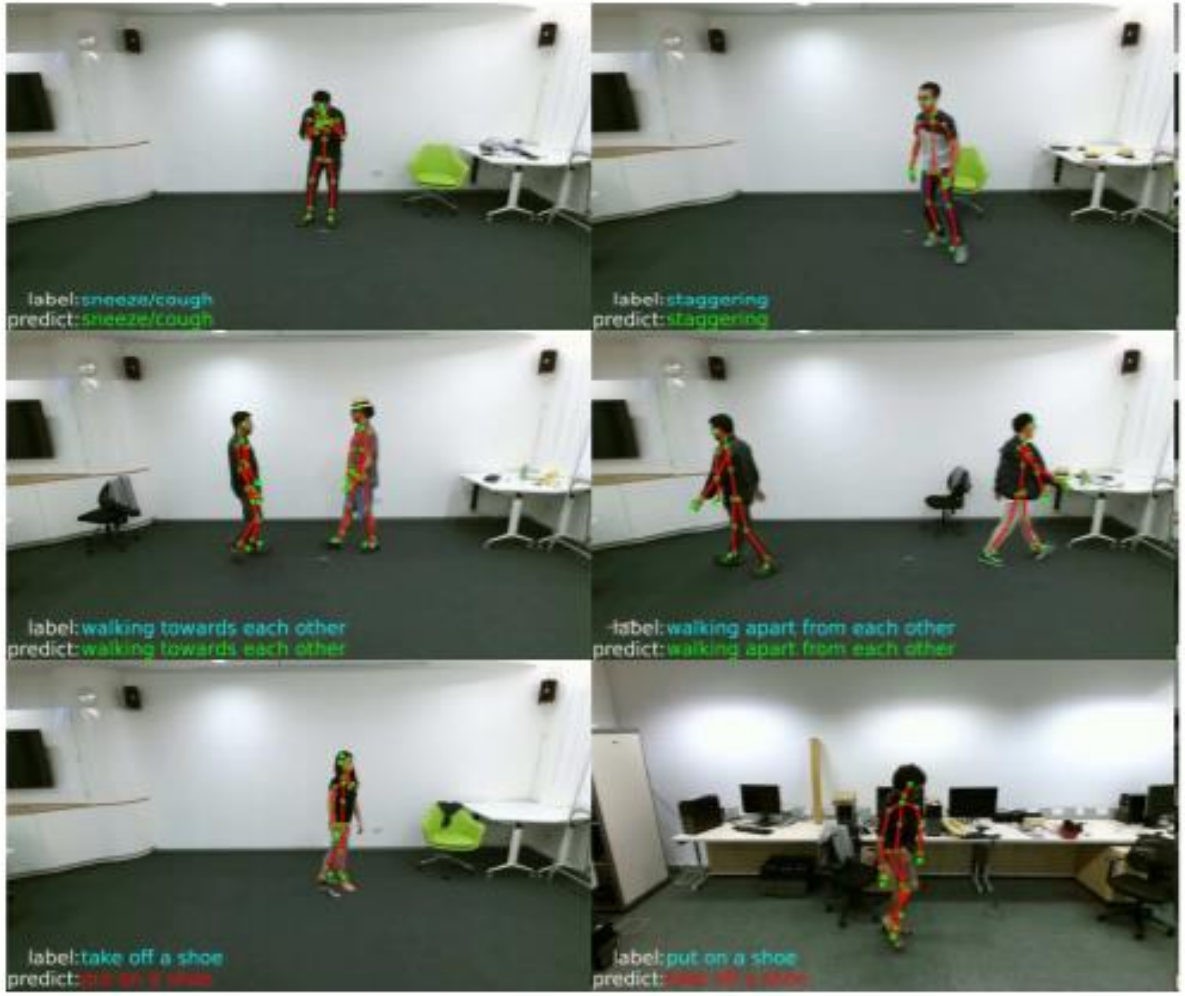
- NTU RGB+D dataset
- Kinect sensor (Microsoft)
  - **Skeleton**
  - **RGB frames**
  - Depth maps
  - IR sequences

- RNN: skeleton
  - 3D coordinates for 25 joints

- CNN: RGB frames

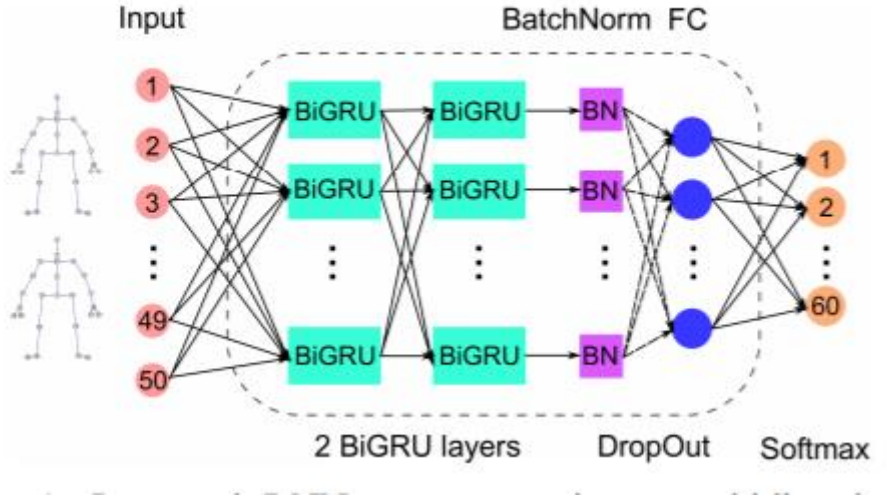
- Multiple subjects, 3 view points

- Evaluation: cross-subject or cross-view
- Training: 2/3 of views or 1/2 of subjects
- Validation: 10% of training



Author's work: Network structure

- RNN: 2 layers of bidirectional GRUs with batch normalization, followed by fully connected layer
  - Also tested multiple combinations with simpler recurrent units of fewer elements
- 3D-CNN: based on state-of-the-art results from Tran et al
  - Fine-tuned after initialization with weights from Sports-1M



Conv1 64 Pool 1	Conv2 128 Pool 2	Conv3 2x256 Pool 3	Conv4 2x512 Pool 4	Conv5 2x512 Pool 5	FC6 4096	FC7 4096	Output 60

Author’s work: Two-stream network

- Two-streamed Network:
  - Both networks process data from the same sample independently, results are then fused

Author’s work: Fusion

- 2 methods

- Decision fusion
  - Majority voting like approach
  - Assign weights to each stream’s decision based on Validation performance, multiply by confidence of decision

- Feature fusion
  - Append outputs of fully connected features from both streams
  - Final classification with SVM
  - Best results

Experiment: Results

- First 8 rows are best previous RNN based results
- Rows 9-16 are author's RNN experiments
- Row 17 is best previous RGB based model
- Rows 18 and 19 are two-streamed network

Nr.	Method	cross subject	cross view
01	Skeleton Quads [2], [9]	38.62%	41.36%
02	Lie Group [2], [10]	50.08%	52.76%
03	FTP Dynamic Skeletons [2], [11]	60.23%	65.22%
04	HBRNN-L [2], [3]	59.07%	63.97%
05	Deep RNN [2]	56.29%	64.09%
06	Deep LSTM [2]	60.69%	67.29%
07	Part-aware LSTM [2]	62.93%	70.27%
08	ST-LSTM (Tree) + Trust Gate [4]	69.2%	77.7%
09	1 Layer RNN	18.74%	20.27%
10	1 Layer LSTM	60.99%	64.68%
11	1 Layer LSTM-BN	64.07%	71.86%
12	1 Layer LSTM-BN-DP	64.69%	73.48%
13	1 Layer GRU-BN-DP	65.21%	70.36%
14	1 Layer BI-GRU-BN-DP	64.78%	73.12%
15	2 Layer BI-GRU-BN-DP	66.21%	72.46%
16	2 Layer BI-GRU-BN-DP-H	70.70%	80.23%
17	3D-CNN [8]	79.75%	83.95%
18	Decision Fusion	82.05%	86.68%
19	Feature Fusion	**83.74%**	**93.65%**

Assessment

- In favor
  - In-depth explanation and experimentation of technical aspects of chosen RNN architecture
  - Strong results in improving performance against current state-of-the-art methods
  - Good presentation of two alternative fusion methods
  - Varied dataset with multiple viewpoints, actions hard to distinguish
  - Confusion matrix shows that mistakes are directly related to similar looking actions – putting on and taking off shoes, clapping and rubbing hands
- Against
  - Not really 3D video – RGB+D data not used, skeleton is more analogous to kinematic data, CNN uses only RGB
  - LSTM-BN-DP performs as well as or better than other single layer RNN architectures, but no 2-layer version presented
  - Unclear how CNN and RNN conciliate the fact that RNN considers entire video but CNN only 16 frames at a time

Relevance to project

- Important alternatives and distinctions in architectures for both spatial and temporal features
- In-depth breakdown of RNN architectures for video-feature extractor
- Alternatives for fusing different types of information
  - In our case, output of CNN would be input to RNN since we have no kinematic data
  - However, we may want to use fusion to include tool labeling information
- Our dataset is even harder

Conclusion

- Solid theoretical background
- Strong results in improving accuracy
- Two-streamed networks are flexible and knowing how to fuse their streams is essential
- Useful for project
- Very technical – no particular application in mind

Reference

- R. Zhao, H. Ali, and P. van der Smargt, “Two-Stream RNN/CNN for Action Recognition in 3D Videos,” arXiv preprint arXiv:1703.09783, 2017
- D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, “Learning spatiotemporal features with 3D convolutional networks,” in 2015 IEEE International Conference on Computer Vision (ICCV). IEEE, 2015, pp. 4489–4497.