ICSPIS 2022
8rd Iranian Conference on Signal Processing and Intelligent Systems
28 – 29 December 2022
University of Science and Technology of Mazandaran, Behshahr, Iran

Adaptive Frame Selection In Two Dimensional Convolutional Neural Network Action Recognition

Alireza Rahnama

Kharazmi University
Tehran, Iran

Alireza Esfahani

University of Science and Technology of Mazandaran
Behshahr, Iran

Azadeh Mansouri

Kharazmi University
Tehran, Iran

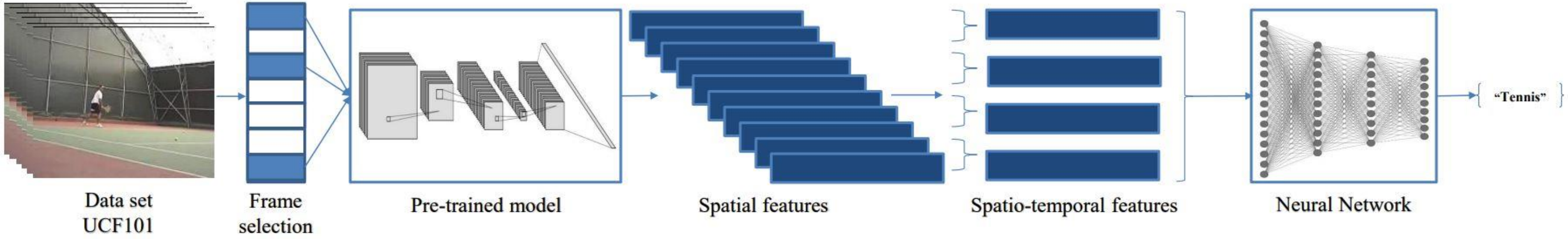
Introduction

- Video is the most important part of this contemporary society:
  - the majority of internet traffic
- Video usage :
  - Action recognition
  - Object detection
  - NLP
- Why do we use frame selection:
  - Redundancy
  - The large volume of data
  - Additional and unusable data
  - Less process

Introduction

An overview of the framework

- Dataset
- Frame-Selection
- Spatial feature extractor
- Temporal feature extractor
- Classification



Algorithm

Adaptive Frame-Selection

1. Read full video
2. Select the first frame
3. FS = Calculate the similarity frame of the last selected frame and the current frame with algorithm[1]
4. SFS = Calculate the average of the similarity frame
5. Check the current SFS with an average[2] of the SFS of all selected frames
6. If SFS $_i$ < [2]Mean of the window:
   - Select the current frame and add SFS $_i$ in the window array
7. Selected frames are used for feature extraction

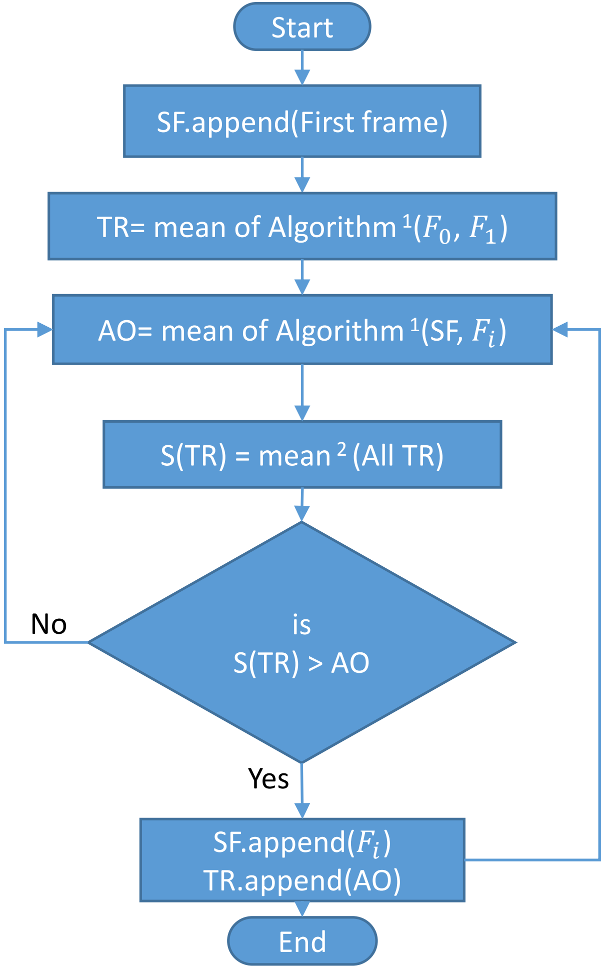
- $^1Frame - Similarity = \frac{2\times F_i\times SF_i}{{F_i}^2+{SF_i}^2+a}$
- $^2\,Mean\ of\ window = \frac{\sum_{i=0}^{n} SFS_i}{n}$

Algorithm

Adaptive Frame-Selection

- $^{1}Frame - Similarity = \frac{2 \times F_i \times SF_i}{F_i{}^2 + SF_i{}^2 + a}$
- $^{2}\,Mean\ of\ window = \frac{\sum_{i=0}^{n} SFS_i}{n}$
- $n = 10$

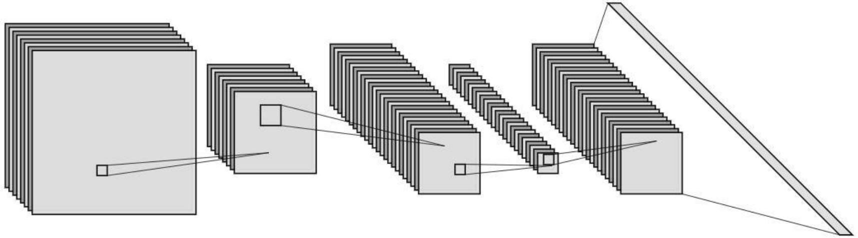
F = Frame
SF = Selected Frame
TR = Temp Result
AO = Algorithm output
S(TR) = scalar of Temp Result


Start
SF.append(First frame)
TR= mean of Algorithm $^1(F_0, F_1)$
AO= mean of Algorithm ^{1}(SF, F_i)
S(TR) = mean ^{2} (All TR)
is
S(TR) > AO
No
Yes
SF.append(F_i)
TR.append(AO)
End

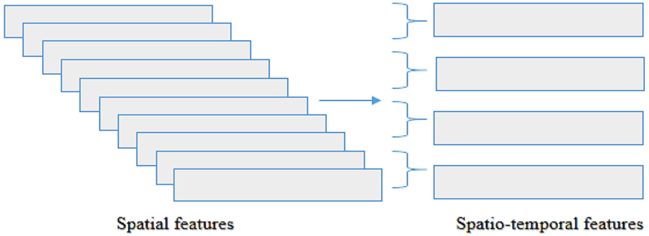
Feature Extraction

Spatio-Temporal pooling

- Spatial features:
  - Transfer learning
  - Pre-trained models:
    - ResNet-50
    - MobileNet

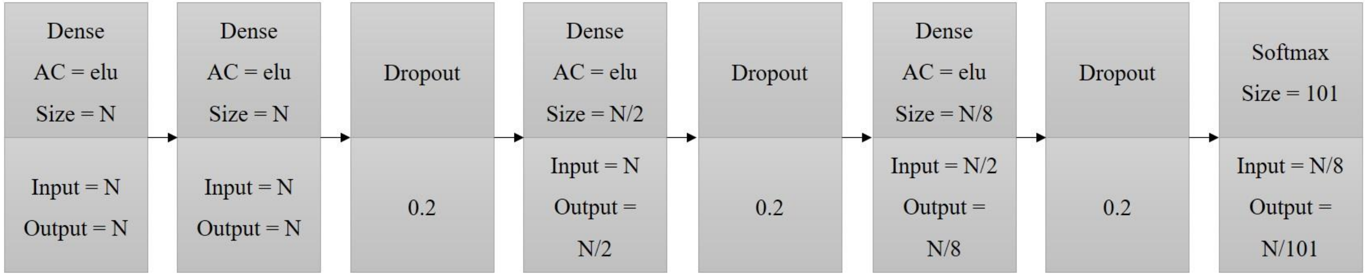


- Temporal pooling
  - Extract the maximum feature of the video
  - Data augmentation



Model

- Layers of Model:
  - Based on the feature vector size

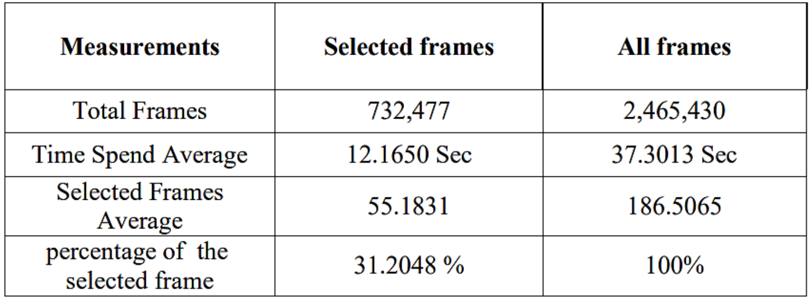


Dense AC = elu Size = N	Dense AC = elu Size = N	Dropout	Dense AC = elu Size = N/2	Dropout	Dense AC = elu Size = N/8	Dropout	Softmax Size = 101
Input = N Output = N	Input = N Output = N	0.2	Input = N Output = N/2	0.2	Input = N/2 Output = N/8	0.2	Input = N/8 Output = N/101

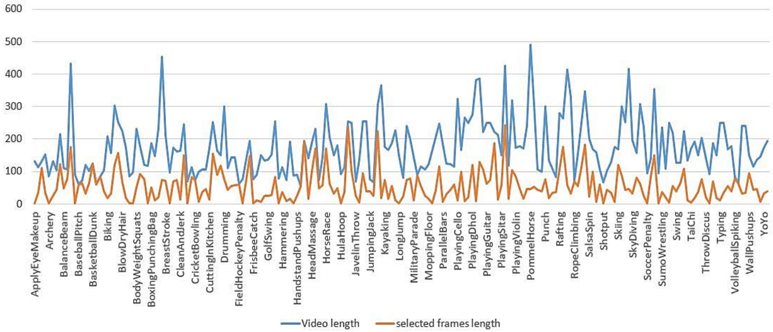
Results

Algorithm

COMPARISON OF TWO SCENARIOS: WITH OR WITHOUT FRAME SELECTION

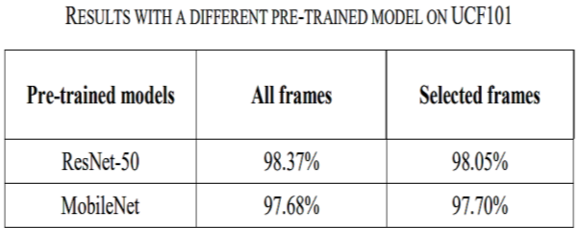


Measurements	**Selected frames**	**All frames**
Total Frames	732,477	2,465,430
Time Spend Average	12.1650 Sec	37.3013 Sec
Selected Frames Average	55.1831	186.5065
percentage of the selected frame	31.2048 %	100%



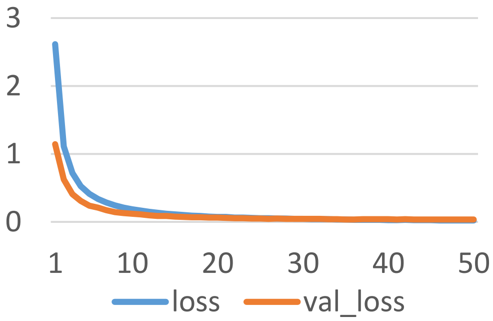
Results

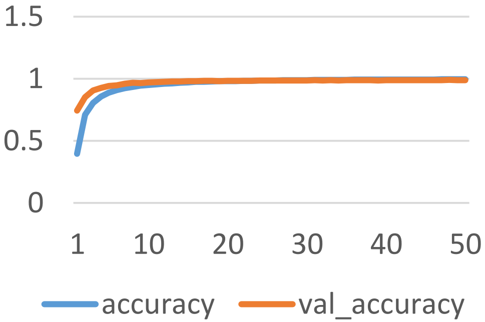
Train&Test



RESULTS WITH A DIFFERENT PRE-TRAINED MODEL ON UCF101

Pre-trained models	All frames	Selected frames
ResNet-50	98.37%	98.05%
MobileNet	97.68%	97.70%



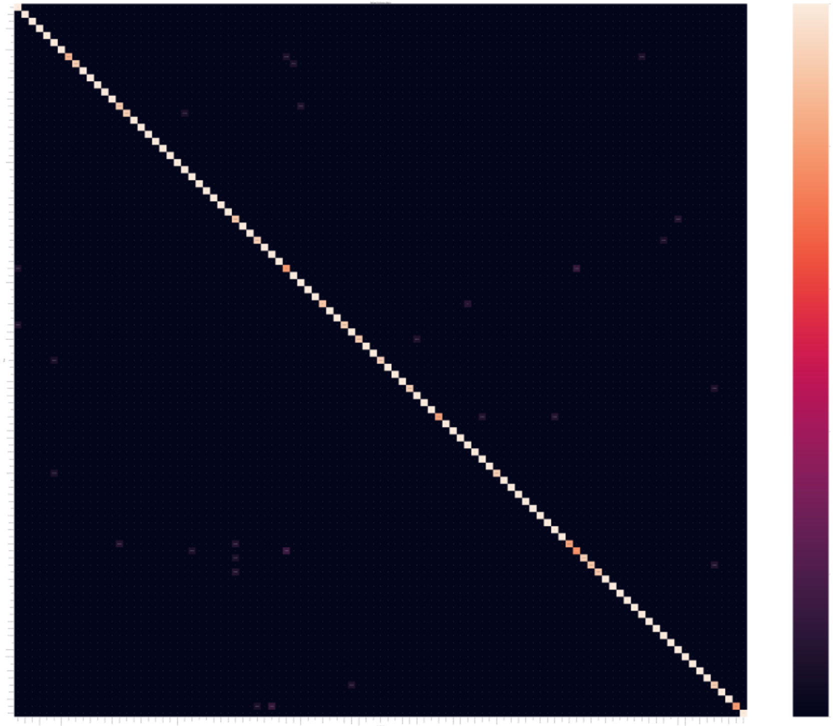


Conclusion and future work

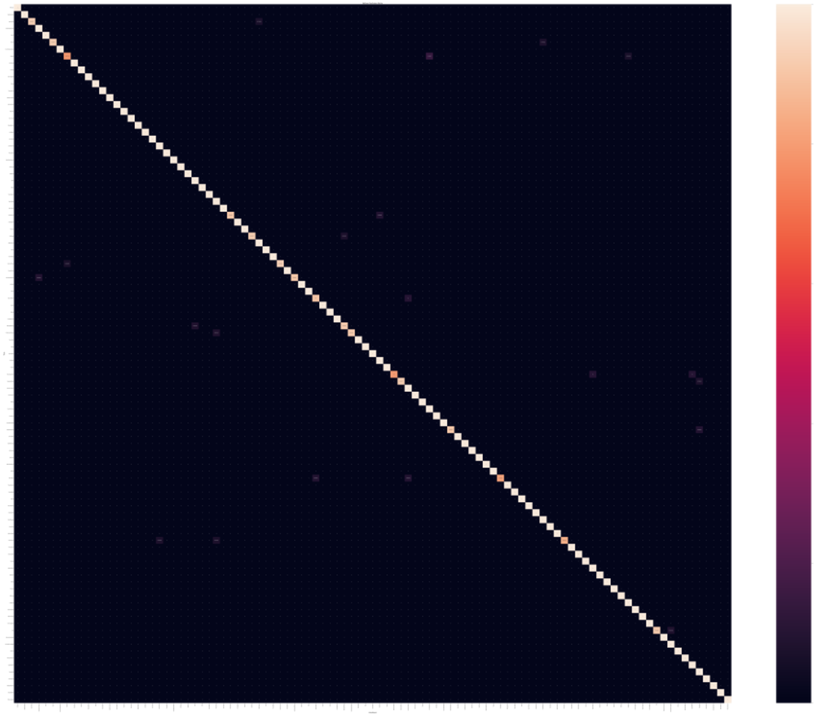
- Process in a shorter time
- Use data in compressed domain
  - Less process
  - Short time

Extra Result

Confusion Matrix



MobileNet



ResNet-50

Thanks for your attention

Email: Alireza.rahnama@khu.ac.ir