Multimodal Action Classifier for Egocentric Vision

Francesco Sorrentino
Politecnico di Torino
s301665@studenti.polito.it

Federico Mustich
Politecnico di Torino
federico.mustich@studenti.polito.it

Giuseppe Atanasio
Politecnico di Torino
s300733@studenti.polito.it

Abstract

This project is about Multimodal Action Recognition, valuable for the Advanced Machine Learning course of Politecnico di Torino. We explored two main datasets: Epic-Kitchens and ActionSense; we observed the importance of temporal aggregation and built our own classifiers for the Action Recognition task. We presented our own implementation of ActionNet and built a new model to exploit CNN over EMG features represented as Mel spectrograms. Finally, we made a Multi-modal action recognition classifier based on E2(GO)MOTION, integrating the new EMG modality to test it on ActionSense. Project code source: https://github.com/Peipi98/aml22-ego

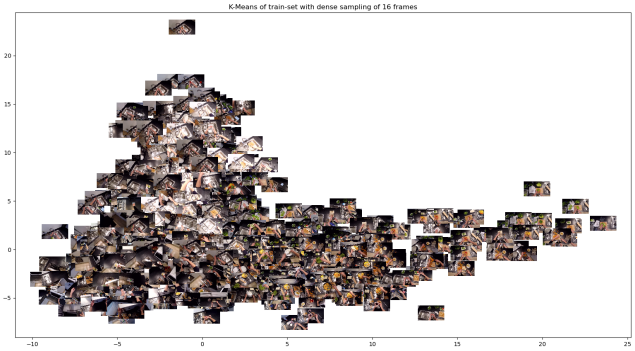

Figure 1. Clustering of EK features, with dense sampling over 16 frames, we can see how all the clips with frame containing specific objects are aggregated: to the right the cutting board, bottom-right double frying pans, from the center to the top a sink, on the left plates/floor/fridge.

1. Introduction

Egocentric vision is an hot topic in research and finds its use in a variety of applications, such as robot manipulation [10], human-object interaction, action recognition [2–6, 8] and others. The research community has argued that using only RGB video data can lead to poor performance on unseen data: in fact this modality is highly biased by the environment, and it may lead to models which try to detect more the object in the scene rather than the action being performed itself [8]. **Temporal aggregation** is a technique to enhance the RGB modality: multiple method were explored by the research community, with different types of sampling and aggregation of frames in order to obtain a better representation of the features [1, 6]. Another possible solution for this task is to use **multi-modal datasets** [5, 7, 9] and **multimodal models**. Using different sensors for data acquisition can lead to better results, since it allows to exploit different feature representations specific to each modality used: for example RGB data can focus on the environmental and visual features while audio data [6] can capture the sound of the action. Different modalities were tested in the past years, like event [5], Electro-Myography (EMG) [9], gaze [9] [7], audio [6] and others.
In this paper we explore the Epic-Kitchens [5] and ActionSense [9] datasets, applying clustering on the features. Exploiting E2(GO)MOTION [4] architecture we extract features with an I3D [5] backbone and then train a classifier with temporal aggregation. Using EMG data, we explore how two different types of feature representations can affect the classification task: on series of EMG readings, we try to provide our own implementation of ActionNet [9] and for image representation of the signal's spectrograms we try to use a simple CNN. Finally we test the integration of the EMG modality into a multimodal action recognition classifier, using both RGB and EMG, from the combined feature extraction to the classification task of the actions.

2. Related Work

Datasets: In literature several examples for datasets depicting recordings of actions from the point-of-view of the person executing them have already been proposed.

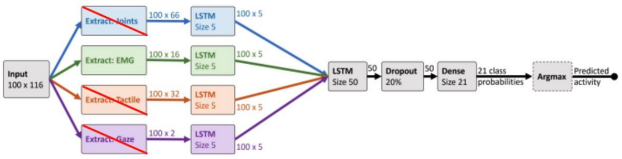

Figure 2. ActionNet model.

Figure 3. Finetuned ActionSense.

These datasets are ideal for research and development of deep learning approaches aimed at solving **First Person (Egocentric) Action Recognition** tasks.
One of the most notable works in this area is the *EPIC-Kitchens* dataset introduced by Damen *et al*. [5]. This dataset contains egocentric videos, in RGB format, recorded by people while performing **non-scripted** daily activities in their kitchens. This provides a unique perspective on human behavior and allows researchers to study fine-grained actions in an actual, realistic, environment. This dataset contains more than 55 hours of video for a total of 39.6k action segments.
DelPreto *et al*. introduced *ActionSense* [9], a multimodal dataset and recording framework for human activities using wearable sensors in a kitchen environment. When compared to EPIC-Kitchens [5], this dataset provides a unique opportunity to study human activities using **multimodal** data recorded from 10 subjects, including multiple video-cameras and microphones recordings, IMU data, finger tracking, tactile sensors data, EMG armband recordings, and eye tracking data, recorded by people while performing scripted activities in a reconstructed kitchen environment, but it has a smaller size of just 12 hours of data. The use of multimodal data allows researchers to study human behavior using multiple sources of information, which can lead to more accurate and robust models.
Other notable datasets include the *Kinetics* dataset introduced by Carreira and Zisserman *et al*. [2], and *Ego4D* introduced by Grauman *et al*. [7], a large-scale egocentric video dataset that contains over 3,000 hours of video recorded by people from all around the globe.

Multimodality Egocentric Action Recognition: Different approaches are possible when trying to fuse different data sources linked to the same performed action.
Kazakos *et al*. focused on multi-modal fusion for egocentric action recognition and proposed *EPIC-Fusion* [6], an architecture for multimodal temporal-binding (i.e, the combination of different modalities within a range of temporal offsets, which could differ one from one another). EPIC-Fusion proposes an architecture relying on three modalities: RGB, Audio and Optical Flow, and combined them to obtain state-of-the-art performance on classification of egocentric tasks on the EPIC-Kitchens [5] dataset.
Similar works include *E2(GO)MOTION*, from Plizzari *et al*. [4], the starting point of our project, that uses, other than RGB, event cameras that captures the pixel-level intensity changes in form of events. In their work they test different multimodal combination using RGB, Flow and Event.
Finally, in Munro and Damen *et al*. [8] they tackled the environmental bias typical of models trained on egocentric action recognition datasets by exploiting the correspondence of different modalities for a self-supervised alignment approach for Unsupervised Domain Adaptation using RGB and Optical Flow modalities.

3. Method

In this section we describe which methods we explored and used for our experiments, from the datasets to the models.

3.1. RGB Features and temporal aggregation

In our first experiment, we decided to explore the RGB features retrieved from a reduced version of the Epic-Kitchens dataset by means of a TRN [1] and of a custom built fully-connected model. Specifically, we focused on just on the subject P8, for a total of 28 videos. Each of these videos had already been divided into individual frames, with each frame measuring 456x256 pixels. To extract the features, we leveraged an I3D backbone pretrained on ImageNet.

We conducted our experiments with two different types of sampling modes:

- **uniform**: selects a specific number of evenly spaced frames in the clip.
- **dense**: takes a number of adjacent frames, possibly spaced by a small stride.

For each type of sampling method, we used a different number of frames (5-10-16-25) per clip, from which we extracted a total of 5 clips for each sample in the dataset. Since we wanted to have a visualization of the trends for each type of sampling, we firstly needed to apply a dimensionality reduction over the features, and in order to do that we considered PCA. Then, we applied K-Means clustering algorithm over the reduced features to assess how the environment affects the distribution of the frames into the clips. (Sec. 5)
Finally, we used the extracted features (at the stage before

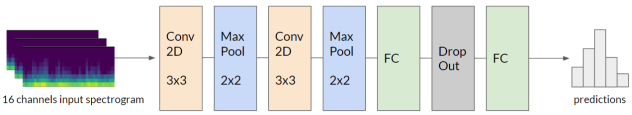

Figure 4. EMG-CNN.

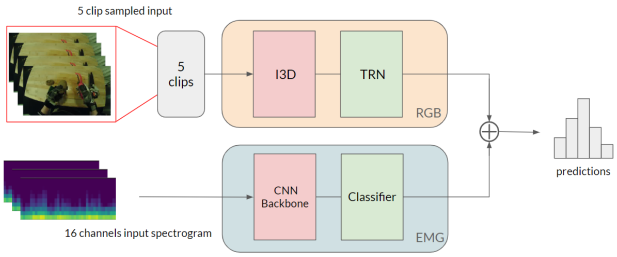

Figure 5. Multimodal Action Classifier.

PCA) to train different classifiers as previously mentioned, with and without temporal aggregation: our goal was to assess the impact that the temporal aggregation over the clips could have had over the correct recognition on the actions.

3.2. Electro-Myography features and ActionNet implementation

In order to improve our study on the action recognition field, we explored ActionSense [9], a multimodal dataset and recording framework which consists of motion, tactile, muscle activity, body tracking, eye tracking, RGB+D video, audio and activity labels.

We considered the EMG (Electro-Myography) data, which have been recorded with Myo Gesture Control Armband from Thalmic Labs, and consists of recordings registerd from two separate devices, one for each of the user's arms, with 8 signal channels for each device recorded at 160Hz and downsampled to 10Hz. We followed the preprocessing used in [9] with one small adjustment: we rectified each channel by taking the absolute value, then we applied a low-pass filter in order to exclude frequencies lower than 5 Hz. After this first step, we applied a joint-normalization for all 8 channels belonging to each armband shifting them in the range [-1,1], using the minimum and maximum values across all channels.

For what concerns the segmentation, we wanted to obtain a series of matrices with fixed lengths to be passed as inputs to the LSTM: as opposed to DelPreto *et al.*, we set a segment duration of 5 seconds in order to generate matrices of length 5 * downsampling_rate=50x16, but for what concerns the remaining part of the segmentation procedure we followed the same process as in DelPreto *et al.*, generating 20 matrices for each activity label within the same experiment, excluding the "None" activity label since our given splits do not have those.

We used both ActionNet as shown in Fig. 2 and its finetuned version proposed by this work (Fig. 3): we excluded the first LSTM, maintaining only the second one that accepts as input matrices 50x16 and size 50, taking the 50 outputs of it and applying a dropout of 20%, a dense layer with the same size as the number of activity labels (20 in spite of 21 since we do not have the None label), and a Softmax instead of an Argmax. The choice to maintain only the second LSTM is due to our suspicions of losing information since the first LSTM of ActionNet has just 5 layers.

3.3. Electro-Myography features and EMG-CNN

In the experiments of DelPetro *et al.* they use an LSTM to train over the sequences of EMG data, as we have done similarly in the previous step. After that we focused on the exploration of a new feature representation for a modality not used in ActionNet: using EMG data as images; We tried to use different types of preprocessing for this task:

- **Spectrogram**: we represent each channel as the repeated SFTF of the signal over a window.
- **Mel Spectrogram**: it is the spectrogram in Mel scale.

Mel spectrograms usually work better with audio signals, since they evenly space the frequency bins in the Mel scale, which approximate the human auditory system better than a normal spectrogram, but from experience it is suggested that it can also help at better representing other type of signals. In the end we choose to report only the mel spectrogram results.
To preprocess signals in the dataset, for each channel we first took the absolute signal, we applied a lowpass filter of 5Hz, then we normalized it between -1 and 1, we cut and padded the signals at 30 seconds to make sure that all the signals were of the same, fixed, length (a requirement for the spectrogram) and finally we computed the Mel spectrogram of the signal.
After we preprocessed each channel, then we stacked them in a 16-channels tensor, 8 for each forearm. This allowed us to feed the features as images of 16 channels into convolutional layers. For this task, we did not want to use an over-complex model, instead we designed a simple CNN. The structure of our EMG-CNN is shown in Fig. 4

3.4. Multimodal action recognition classifier

After we experimented with the two modalities (RGB and EMG), we built our multimodal classifier using RGB and EMG features of ActionSense. For this task we used only the video S04_1. Each video was too heavy to be

downloaded and sampled, so we had to settle with using only one. This implied that **we only had 51 training samples and 8 test samples**. The main issue was that, considering only verb classes, we have a minimum of 11 classes, so the test samples did not covered all the classes. To partially solve this issue we tried to **resample the data to augment the dataset, so we cut or padded every sample into 5 seconds splits**: unluckily this resulted in a overall degradation of the accuracy during pretraining, probably for a combination of a loss of information (considered time windows passed from 30s to 5s) and a poorly designed network, so we did not test it on the multimodal network and it is left as a future work. As in Fig. 5 the network is composed of an RGB stream where we used the same pipeline as we did for Epic-Kitchens, so after taking 5 clips of n frames with dense or uniform sampling, we extracted the features with a I3D backbone (pretrained on imageNet) and then we fed them into the TRN classifier; for the EMG stream we used the EMG-CNN as a backbone (pretrained on the whole available ActionSense dataset) and then we fed the features into a simple classifier composed of 2 fully connected layers. The accuracy has been calculated on the sum of the two scores.

4. Experiments

In this section we explore all the experiments that we made, from feature clustering to the multimodal action recognition classifier. All these experiments were run on our personal computers, with and without GPUs, and some training was made on the free version of Google Colab, where we had limited resources and time. When we used video as inputs, we used the frames to first extract the features and then to train a classifier. For ActionSense, since each video was over 20 GB in size, we only used one video (S04_1): that means only 59 sample (51 training, 8 test). **That is enough to test whether the complete pipeline works correctly, but it is not sufficient to assess the performance of the multimodal classifier and compare it with to the other two single modalities.** For completeness we reported the results and ran an ablation study on the model, but for the above reasons they are not to be intended as reliable.

4.1. Clustering RGB Features

As discussed in Sec. 3.1 we extracted features using different combination of kind of sampling and number of frames, then we applied PCA (number of components = 2) and finally we applied K-Means (from Fig. 7 to Fig. 14). We observed that uniform and dense sampling modes tend to process data which are then clustered in different ways by the K-means algorithm. For instance, in Fig. 9, considering that each frame visualized on the plot is the central one for one clip of the feature, we can observe that the features are distributed depending on the objects in the images, in fact on the right we can see a concentration of frame with cutting boards, on the top the ones which have sinks and on the left there are background objects (fridge, floor etc..).
By a quick look at both dense and uniform sampling clustering plots in Sec. 5, we can see that while in the dense plots it is easier to discern the features that has frame with different object, for instance sinks and frying pans, in the uniform ones it is much confused, it seems that it focuses much more on sinks. We suppose the problem is due to the fact that the frames sampled using dense sampling are closer to the central frame unlike those obtained with uniform sampling.

4.2. TRN vs Fully Connected

To test the two classifiers, we trained them both for 5000 iterations with simulated batch size, using a learning rate of 0.001, weight decay of 1e-7 and a momentum of 0.9, and we used Cross Entropy Loss. The Fully Connected classifier was tested feeding one clip at time into the classifier, instead for the TRN we fed all 5 clips at once because of the temporal aggregation that is done into it.

Sampling	Frames	Acc@1	Acc@5
Dense	5	56.32	96.09
	10	57.70	96.55
	16	58.62	95.86
	25	57.01	96.32
Uniform	5	48.05	94.25
	10	50.80	96.09
	16	52.87	95.40
	25	54.71	95.40

Table 1. TRN: Top1 and Top5 accuracies

Sampling	Frames	Acc@1	Acc@5
Dense	5	54.02	98.16
	10	55.40	98.62
	16	57.70	98.39
	25	57.93	98.39
Uniform	5	48.97	94.94
	10	49.89	97.01
	16	52.64	98.39
	25	53.10	96.55

Table 2. FC: Top1 and Top5 accuracies

As in Tab. 1 and Tab. 2, the dense sampling seems to perform better with these two specific classifiers, but we can also see how temporal aggregation in TRN returns better results in each of the two sampling modes. Even if the uniform sampling mode allows for a better view of the action in the clip, the classifiers works better with the dense mode

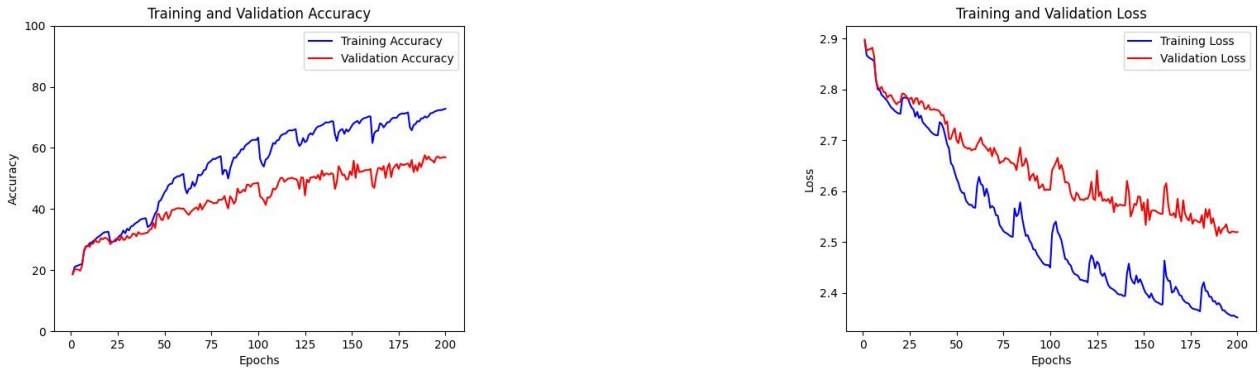

Figure 6. Accuracy and loss curves of our model, with 5 seconds configuration.

and we can assume that it is because of the similarities in the frames of each clip, since they are closer in time.

4.3. ActionNet vs Finetuned version

Starting from Sec. 3.2, we tested both ActionNet and our fine tuned version of it in order for us to have a comparison of the performances. We wanted to focus on the differences between ActionNet and our model, and also on how giving as input either 10s or 5s feature matrices impact on the classification accuracy.

According to Tab. 3, **we observed that the finetuned model performs slightly better than ActionNet with respect to Acc@1 and Acc@5 for both 5s and 10s configurations**. These results confirm our hypothesis on the temporal propagation of the data: the LSTM in our model maintains longer temporal informations since we only consider the second LSTM with 50 layers, as opposed to ActionNet which has a first LSTM with 5 layers, used to aggregate the multimodal features, and therefore it tends to lose much of the temporal information. For the sake of the scientific exploration, we must say that our model is more **prone to overfitting** when compared to ActionNet. About this issue we can hyphotize that it could be due to the poor size of the dataset over which we have tested the models and it could benefit from being tested over the entire ActionSense dataset. Another explanation could be the excessive simplicity of our model architecture. For what concerns the dimensions of the feature matrices, we can observe that 5 seconds crops perform slightly better than their 10 seconds counterpart.

4.4. EMG: CNN

To test our EMG-CNN (Fig. 4) we first preprocessed the signals as in Sec. 3.3. The model has been trained using Adam optimizer with a learning rate of 0.0001, for 20 epochs, 32 batch size, PolynomialLR scheduler with iteration equals to epochs and Cross Entropy Loss. We tested the model both on verbs and complete actions, and the first one was selected as pretrained backbone for the multimodal experiment.

	5 seconds		10 seconds	
	Acc@1	Acc@5	Acc@1	Acc@5
ActionNet	39.82	60.18	32.11	60.22
Our Model	55.82	75.64	50.22	73.56

Table 3. Top 1 and top 5 validation accuracies of activity labels over 200 epochs, using Cross Entropy Loss, Adam optimizer with learning rate=1e-3 and Cosine Annealing with Warm Restarts each 20 epochs.

	Acc@1	Classes
EMG-CNN	70%	11 (verbs)
EMG-CNN	55%	20 (actions)

Table 4. Top1 Accuracy on ActionSense

As reported in Tab. 4 the top-1 accuracy on ActionSense (using the entire dataset) over the actions is close to the results of Tab. 3, so with this configuration **we cannot confirm an improvement on the measured performance.** We previously mentioned problems on testing the multimodal network given the small size of the dataset, so we tried resampling the dataset into 5 seconds readings, padding one record or splitting it in multiple record of 5s. However this led to worse results. As a matter of fact, the model struggled to learn from the features, obtaining an accuracy of just 28% on verbs. For a future work it could be interesting to test different models and architectures more suitable to EMG features.

4.5. Multimodal Action Classifier

Our Multimodal Classifier was trained on EMG and video recordings obtained from S04_1 of ActionSense, which contained about one hour of data, from which we extracted each frame in 456x256 px size. The model has been trained on 350 iterations with a batch size of 32 samples. We sampled the RGB data either in uniform or dense mode with 5-10-16-25 frames.

On a first look, the obtained experimental results (Tab. 5) may seem promising but, sadly, **the available data was very poor in quantity**. It only counted 59 total samples belonging to 11 different and unbalanced classes (considering only the verbs and not the complete actions). This severely affected the meaningfulness of the measured performance of the classifier on the test set, which counted a mere total of 8 samples.

Sampling	Frames	Acc@1	Acc@5
Dense	5	75.00	87.50
	10	75.00	87.50
	16	75.00	87.50
	25	75.00	87.50
Uniform	5	62.50	87.50
	10	75.00	87.50
	16	75.00	87.50
	25	75.00	87.50

Table 5. Multimodal RGB+EMG: Top1 and Top5 accuracies

4.5.1 Ablation Studies

We wanted to assess the contribution of each modality to the overall performance of the classifier. In order to do so we retrained the classifier by ignoring either the EMG or the RGB models. The obtained results are those of Tab. 6 and Tab. 7.

Sampling	Frames	Acc@1	Acc@5
Dense	5	87.50	87.50
	10	87.50	87.50
	16	87.50	87.50
	25	87.50	87.50
Uniform	5	75.00	87.50
	10	62.50	75.00
	16	75.00	87.50
	25	87.50	87.50

Table 6. RGB-only: Top1 and Top5 accuracies

From the results it would seem that the best performance is reached when using the RGB classifier alone and that the contribution from the EMG one is only detrimental.

Sampling	Frames	Acc@1	Acc@5
Dense	5	50.00	87.50
	10	50.00	87.50
	16	50.00	87.50
	25	50.00	87.50
Uniform	5	50.00	87.50
	10	50.00	87.50
	16	50.00	87.50
	25	50.00	87.50

Table 7. EMG-only: Top1 and Top5 accuracies

However, **the size of the training and test sets is probably affecting these results**. With a test sample of such small size the above results are not to be considered as definitive results and the effective performance of the multimodal classifier has yet to be assessed.

Nonetheless, even though the results are unreliable, they suggest that the RGB classifier performs better on dense sampling mode as opposed to the uniform one.

5. Conclusions

In this work we analyzed the performances of TRN and Fully Connected models for RGB features only, with uniform and dense temporal aggregations in terms of accuracy and cluster distribution, with 5, 10, 16 and 25 frames, and observed that dense performs better than uniform (especially with 10 frames configuration). We also compared ActionNet with its finetuned version proposed in this paper over 5 and 10 seconds crops of the EMG readings, and observed that both the latter and the 5s crops perform better. We have tested a multimodal with RGB and EMG (as Mel spectrograms) and compared the results with the single relative models, observing that RGB-only seems to perform better than the multimodal one, even if the small size of the training data must be take into account. Lastly, the EMG-CNN model performs poorly when compared to the others in top 1 accuracy.

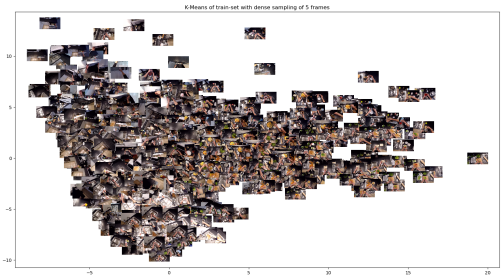

Figure 7. K-Means cluster for Dense Sampling (5 Frames per clip)

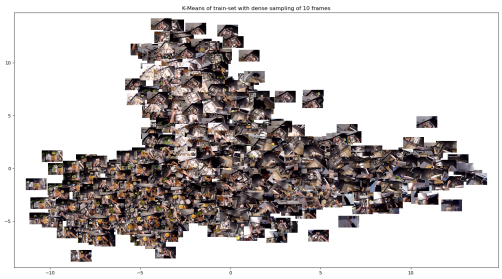

Figure 8. K-Means cluster for Dense Sampling (10 Frames per clip)

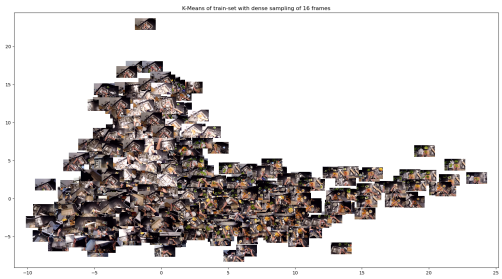

Figure 9. K-Means cluster for Dense Sampling (16 Frames per clip)

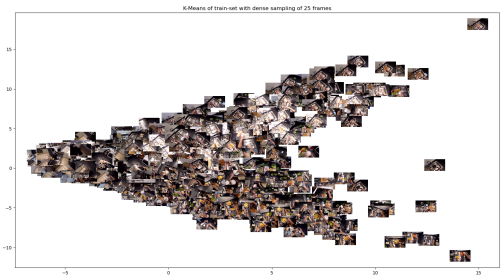

Figure 10. K-Means cluster for Dense Sampling (25 Frames per clip)

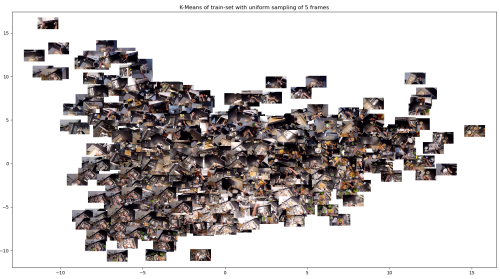

Figure 11. K-Means cluster for Uniform Sampling (5 Frames per clip)

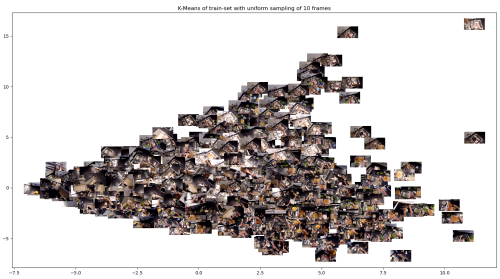

Figure 12. K-Means cluster for Uniform Sampling (10 Frames per clip)

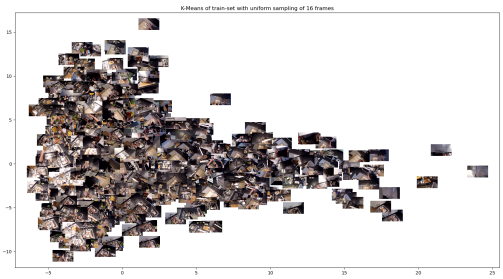

Figure 13. K-Means cluster for Uniform Sampling (16 Frames per clip)

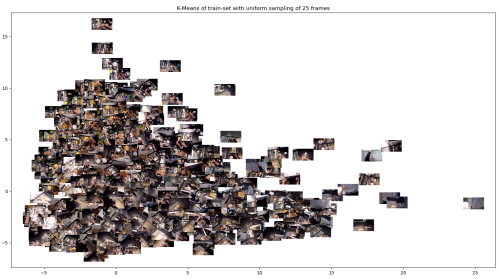

Figure 14. K-Means cluster for Uniform Sampling (25 Frames per clip)

References

[1] Aude Oliva Antonio Torralba Bolei Zhou, Alex Andonian. *Temporal Relational Reasoning in Videos*. 2018. 1, 2

[2] J. Carreira and A. Zisserman. *Quo vadis, action recognition? A new model and the kinetics dataset*, 2017. 1, 2

[3] C.-F. Chen, R. Panda, K. Ramakrishnan, R. Feris, J. Cohn, A. Oliva, and Q. Fan. *Deep analysis of cnn-based spatio-temporal representations for action recognition*, 2020. 1

[4] G.Goletto M.Cannici E.Gusso M.Matteucci B.Caputo C.Plizzari, M.Planamente. *E^2(Go)motion: Motion augmented event stream for egocentric action recognition*, 2021. 1, 2

[5] D. Damen, H. Doughty, G. M. Farinella, S. Fidler, A. Furnari, E. Kazakos, D. Moltisanti, J. Munro, T. Perrett, W. Price, and M. Wray. *Scaling egocentric vision: The epic-kitchens dataset*, 2018. 1, 2

[6] A.Zisserman D.Damen E.Kazakos, A.Nagrani. *Epic-fusion: Audio-visual temporal binding for egocentric action recognition*, 2019. 1, 2

[7] K. Grauman, A. Westbury, E. Byrne, Z. Chavis, A. Furnari, R. Girdhar, J. Hamburger, H. Jiang, M. Liu, X. Liu, M. Martin, T. Nagarajan, I. Radosavovic, S.K Ramakrishnan, F.Ryan, and M.Xu E.Z.Xu J.Malik J.Sharma, M.Wray. *Ego4d: Around the world in 3,000 hours of egocentric video*, 2021. 1, 2

[8] D.Damen J.Munro. *Multi-modal domain adaptation for fine-grained action recognition*, 2020. 1, 2

[9] Yiyue Luo Michael Foshey Yunzhu Li Antonio Torralba Wojciech Matusik Daniela Rus Joseph DelPreto, Chao Liu. *ActionSense: A Multimodal Dataset and Recording Framework for Human Activities Using Wearable Sensors in a Kitchen Environment*, 2022. 1, 2, 3

[10] V.Kumar C.Finn A.Gupta S.Nair, A.Rajeswaran. *R3m: A universal visual representation for robot manipulation* , 2022. 1