ELL888 - Assignment 2

Chahat Chawla(2016MT10492), Nishad Singhi(2016EE10107), Hritik Bansal(2016EE10071), Gantavya Bhatt(2016EE10694)

1 Problem Statement

Data comprises frames of NPTEL videos, taken in a digital screen setting where mode of scribing is on digital sheets or digitally on prepared slides. We wish to extract only those frames which are maximally filled and are optimal for note taking. We cast this as a supervised learning problem where X is the frame and Y is 0/1 label, stated otherwise per frame binary classification problem.

2 Data Visualization

We observe the first and higher order temporal gradients of the frame sequence and its correlation with the labels, which we find to be realated. We use optical flow and gradient information as one of the channel to our learning algorithms.

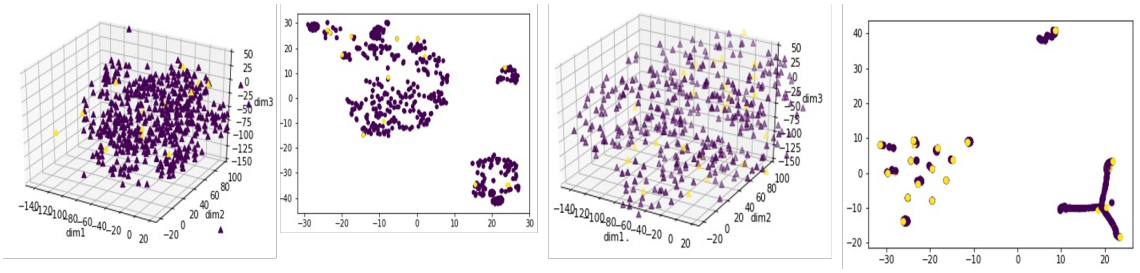

Figure 1: tSNE plots in 3D and 2D for two different lecture settings(handwritten and slides mode)

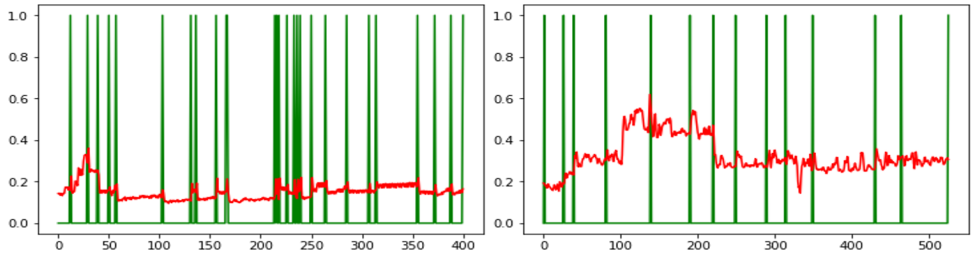

Figure 2: First Order(a) and Higher Order(b) Temporal Gradients (red) and labels (green)

3 Stepwise analysis of the problem

- As a baseline step, we try **vanilla CNN** where each frame is casted as one data point. **An motivation from Cognition:** Human decisions can be interpreted being instantaneous when practically taking notes.
- In the above spirit, we apply **t-SNE** to the data and observe the data manifolds in the two and three dimension. We try **SVM** as well, which fails on cross-validation due to **hihgly entangled classes in the data** as evident from t-SNE.
- Across lectures, there is huge variation in the data distribution and nature of the frames. There is heavy imbalance between class 1 and class 0 for any lecture which poses difficulty in learning.
- We try taking **temporal gradients and higher order temporal derivatives**, for inspection as well as a feature for the learning algorithms. We conclude that Y = 1 implies difference exceeds hypertuned threshold.
- By manual inspection of the data, we realise two essential parameters of classification, extent and location of the text and the location of the hand of the professor.
- Observing lectures where instructor is writing on sheets, another proposal of the problem can be localizing the hand in the right most corner ensuring that the page is filled to a particular threshold.
- Moving from our baseline CNN, we assume **Markov property** in the frames and test one step **Markovian model** and extend it to **k-step Markov model**.
- This means that if the last sheet or the previous sheets are near to completion, it implies following pages are highly probable to be maximally filled.
- An ideal situation would have been solved by this, but we realise that there are times when professor jumps back and forth in the sequence.Also, in the slides mode, there are datapoints where there is long term dependency.
- Innovating in the above direction, we try **one step(and k step) bidirectional Markovian** assumptions.
- If the next page suddenly goes blank(or equivalently there are major changes), it is a good indicator of Y = 1.
- Before jumping to models that incorporate long term dependencies in the data, we try **Context model**, taking inspiration from Natural Language Processing. We feed the **context of an frame**, which is hypertuned, to a CNN model.
- Finally, we try **bidirectional LSTM** models. Owing to the data being imagery, we consider two convolutional LSTM models, one where **CNNs are used for feature extraction in simultaneous training with the LSTM model**, and the other one being the **vanilla CNN LSTM model where we replace the operations in LSTM cell by the convolutional** counterparts.

4 Feature Aiding

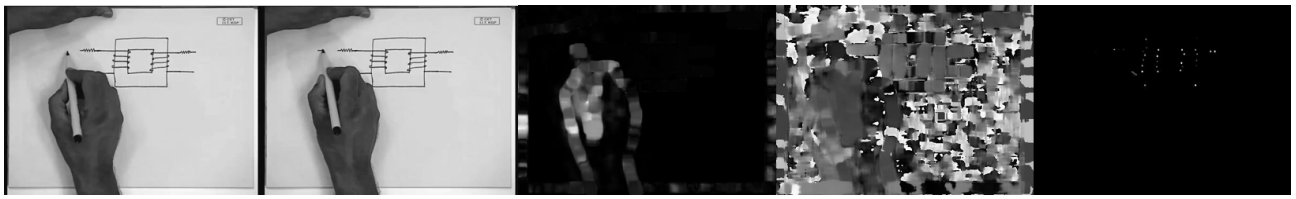

Figure 3: Consequtive Frames(a) & (b), Dense Optical Flow Magnitude(c) and Phase(d), Lucas Kanade Optical Flow(e)

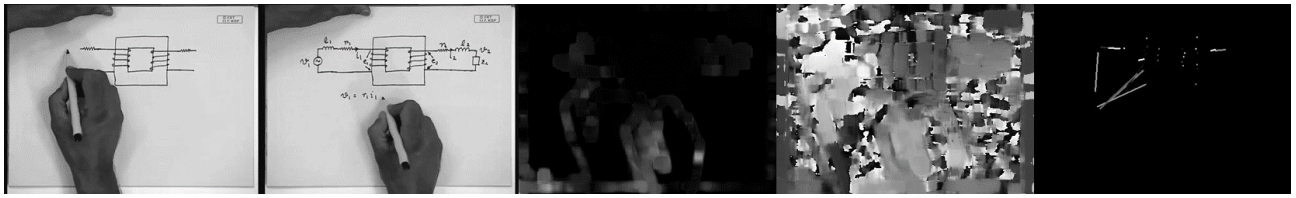

Figure 4: Distant Frames(a) & (b), Dense Optical Flow Magnitude(c) and Phase(d), Lucas Kanade Optical Flow(e)

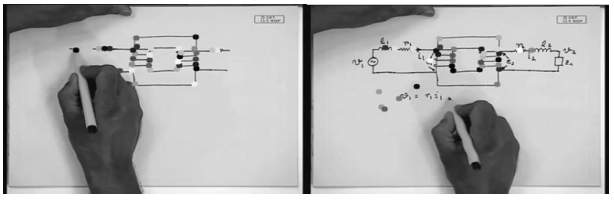

Figure 5: Shi Tomasi Corner Points under tracking

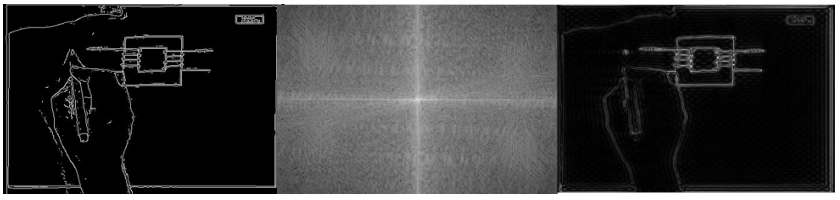

Figure 6: Canny Edge Detection(a) Fourier Transform Phase(b) High Pass filtering as edge detection(c)

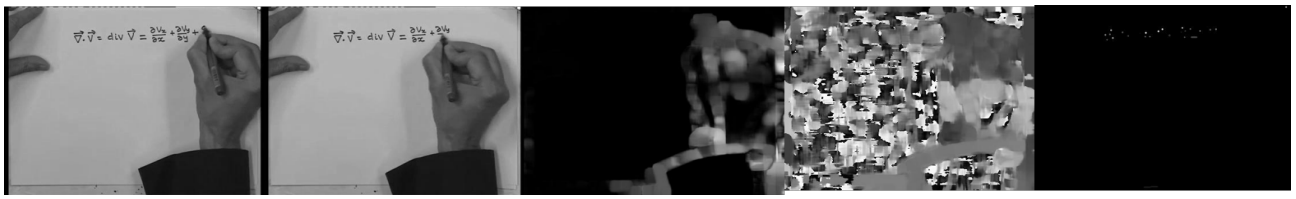

Figure 7: Consequtive Frames(a) & (b), Dense Optical Flow Magnitude(c) and Phase(d), Lucas Kanade Optical Flow(e)

Figure 8: Distant Frames(a) & (b), Dense Optical Flow Magnitude(c) and Phase(d), Lucas Kanade Optical Flow(e)

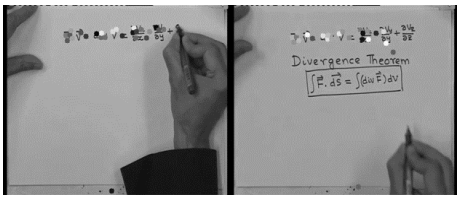

Figure 9: Shi Tomasi Corner Points under tracking

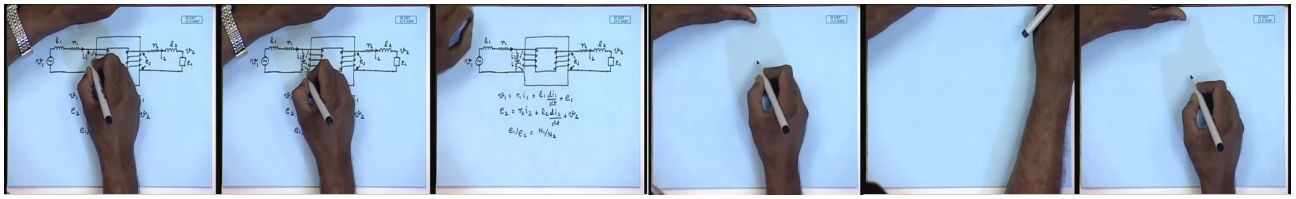

Figure 10: Context of an image

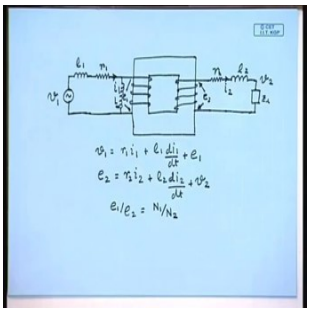

Figure 11: Context of an image

Since motion of the hand of the professor and the extent and location of the data on the screen are two natural features, we apply Signal Processing and classical Computer Vision techinques to aid the learning algorithm. We address these by –
1)Canny edge detection is applied to extract the hands and text from the frames.
2)We take the first frame, detect some **Shi-Tomasi corner points**(along with some other good features to track) in it, then we **iteratively track** those points using **Lucas-Kanade optical flow**.
3)Dense Optical Flow Computation: We compute the optical flow for all the points in the frame. It is based on **Gunner Farneback algorithm** which is explained in **Two-Frame Motion Estimation Based on Polynomial Expansion** by Gunner Farneback in 2003. We get a 2-channel array with optical flow vectors, (u,v). We find their magnitude and direction.
4)We use **Fourier transforms** also as an feature. We utilize **High pass filtering** as an **edge detection operation.**

5 Architectures

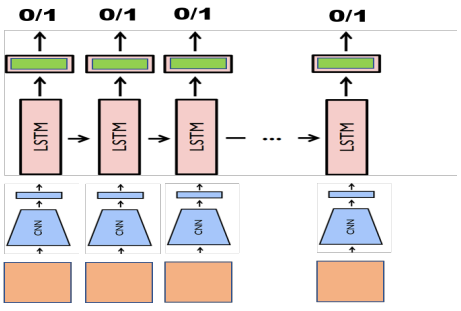

Figure 12: LSTM with features from CNN

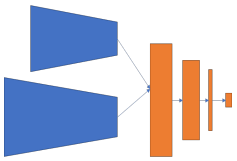

Figure 13: Contextual CNN (inspired from NLP)

6 Results

1) Support Vector Machines Confusion Matrix

$$Confusion = \begin{bmatrix} 1162 & 1 \\ 97 & 60 \end{bmatrix}$$

Though we see inclincation towards predicting 0, but we do get fair number of label 1 for 1 class examples.
2) Vanilla CNN for classification doesn't perform well or get good F1 score/precision metrics owing to the skew in the data.
3) Loss curves for contextual CNN and the CNN-LSTM over the training phase. We observe noisy learning due to high variation in the data distribution.

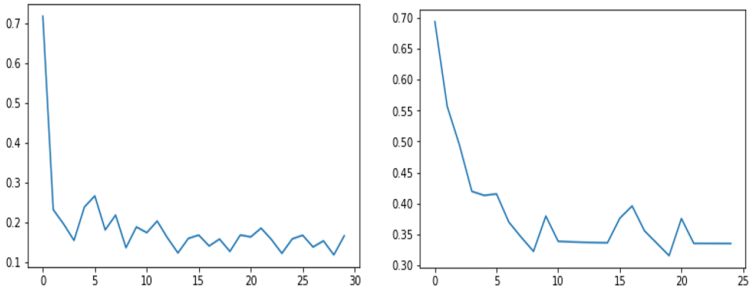

Figure 14: Loss for Contextual CNN and CNN-LSTM over training

Model	Performance
SVM	Accuracy 97.83 , (Confusion Above)
CNNs	98.60 Accuracy
Contextual CNN	55.18 F1 SCORE
LSTM with features from CNN	56.7 F1 SCORE
Convoluted LSTM	53.2 F1 SCORE

Table 1: Evaluation Metrics

Note: Owing to the data skew, most of the learning algorithms collapsed to near 95 percent accuracy and poor F1 score. We recognise these cases as when the algorithm is not learning anything(or properly). Therefore, we train models to make sure that the algorithm learns something, improving F1 Score which is the key measure for us here, at the cost of drop of accuracy.F1 Scores for the models where our algorithm is learning something(evaluated by manual inspection on actually label one samples) , have been listed in the table above.

7 Implementation Notes

1)Due to variation in frame size, we downscale frames with Anti-Aliasing.
2)We normalize each image by its maximum.
3) We test different activation functions and batch normalization.
4)We use Adam for optimization.

8 Learnings from the assignment

- Implementing sequential models was an interesting and involved exercise.
- Deep Learning is a highly empirical process and requires wide.
- Although Deep learning provides for rich representational learning, feature engineering can help tremendously when working with less data, again remarking data hunger of Deep Learning.
- Though neural networks can asymptotically learn everything, but it is always better to explicitly impose structures that are known. For this reason, we impose first order Markov assumption and evaluate them.

9 Improvements

The problem admits three further experimentations.
1)Generating modelling(preferably GANs), to aim for generative behaviour and at the same time improving predictive behaviour of the current time frame. We left this approach owing to the difficulty in being able to achieve equilibrium with Video data.
2)HMM models to explicitly impose the Markovian property[1].Though NN sequence models can learn Markov property, but practically when there is Markov behavior, HMM have been found to work well with the limited data compared to the Deep Learning techniques.
3)Attention Models as they are proven efficient memory mechanisms in sequence learning problems.

References

[1] Comparing Hidden Markov Models and Long Short Term Memory Neural Networks for Learning Action Representations
https://pub.uni-bielefeld.de/download/2903474/2907910/MOD2016.pdf

[2] http://www.diva-portal.org/smash/get/diva2:273847/FULLTEXT01.pdf