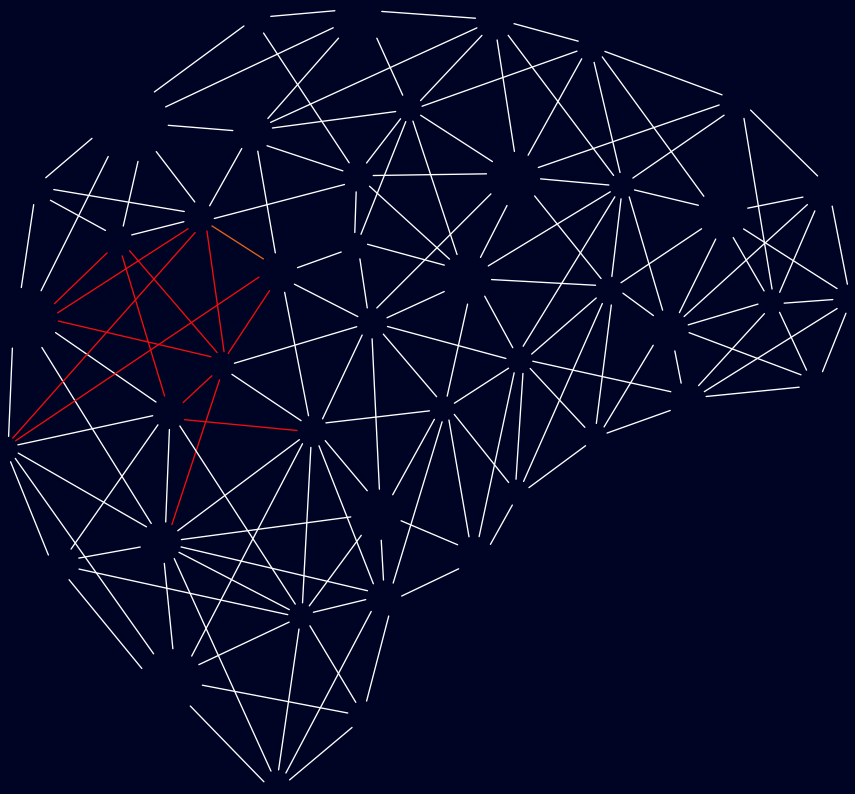
SURPRISE



Mechanisms of Surprise in Artificial Neural Networks

Proposal for research by David Brust

Table of Contents

Introduction

This proposal serves to prompt interest in the research of 'surprise' mechanisms within Deep Neural Networks (DNNs). The proposal will follow a *proof of concept* for the creation of a simplified DNN architecture and rudimentary analysis of results from 'surprise' data in a DNN network to show the validity of the proposal.

Modern DNNs have provided the best model for similarity to the natural brain, especially regarding visual representations (Khaligh-Razavi et al. 2014). The prospect of using an *in-silico* approach to neuroscience research, and its ease in studyies–due to its speed and repeatability–has shown promise in allowing broader analysis of neural computation. While DNNs do not approximate all representational ability of the brain (Baker et al. 2018), they do retain parallels in their ability to produce brain-like representations and may be able to provide a functional quantitative model for many aspects of biological computation (Zhuang et al. 2021). DNN research is useful in providing a better understanding of the similarity of DNNs to brain regions and provides a way to develop new tools for the study of natural cognition. The understanding of the ability for DNNs to parallel human cognitive ability is a cornerstone of *in-silico* research.

A type of DNN model known as a Convolutional Neural Network (CNN) shows a particular ability in visual representation and recognition due to the emergence of object detectors in training (Zhou et al. 2015). However CNN networks, which traditionally use a feed forward mechanism, do not retain sensitivity to sequences in visual stimuli–like in frames of a video. The traditional way to address this shortcoming–especially in the case of video data–is a three dimensional convolution, housing all frames within the input. (Ji et al. 2013). This, however, leaves the network processing data without the element of time. Instead a network that works on data streams would be better utilized for this proposal.

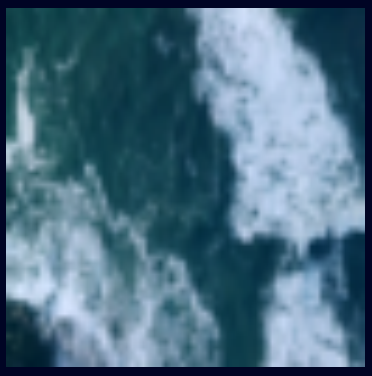
In light of this, the DNN used for to process the 'surprise' video streams uses two sequences of training. First a simple unsupervised convolutional Auto-encoder network is trained to reduce input frame (single image) data through an encoding process, then decode the reduced data to reproduce the original frame through a process of decoding. After training the encoder is used as the first layer for a Long Short Term Memory (LSTM) network. This Network has internal states that change based on data sequences, that are fed back into the network for the next iteration in data input. Using this network sequence, both the advantages of convolutional object detection and time sequence sensitivity of an LSTM can be used. This allows for classification of video sequences not only based on the visual data within each frame, but the order and sequences that the frames are presented.

Training Data

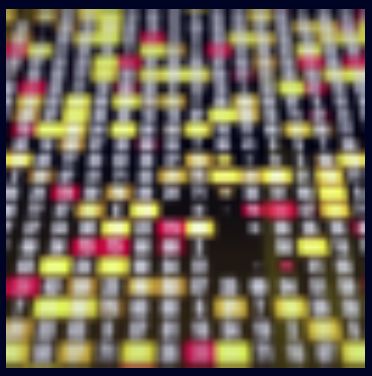
City

Forest

Ocean



Stock



The network was trained on 4 original videos–City, Forest, Ocean, and Stock– which show New York city streets, a pan through of a forest, an overhead video of ocean waves, and a stock board with flashing indicators of different colors.

For City, the video pans right across a street in New York City. Pedestrians and cars pass while large billboards brightly flash advertisements.

For Forest, the video moved forward through the trees of a forest. The frames are mostly green or brown in color, with the difference consisting mainly of shade and light value.

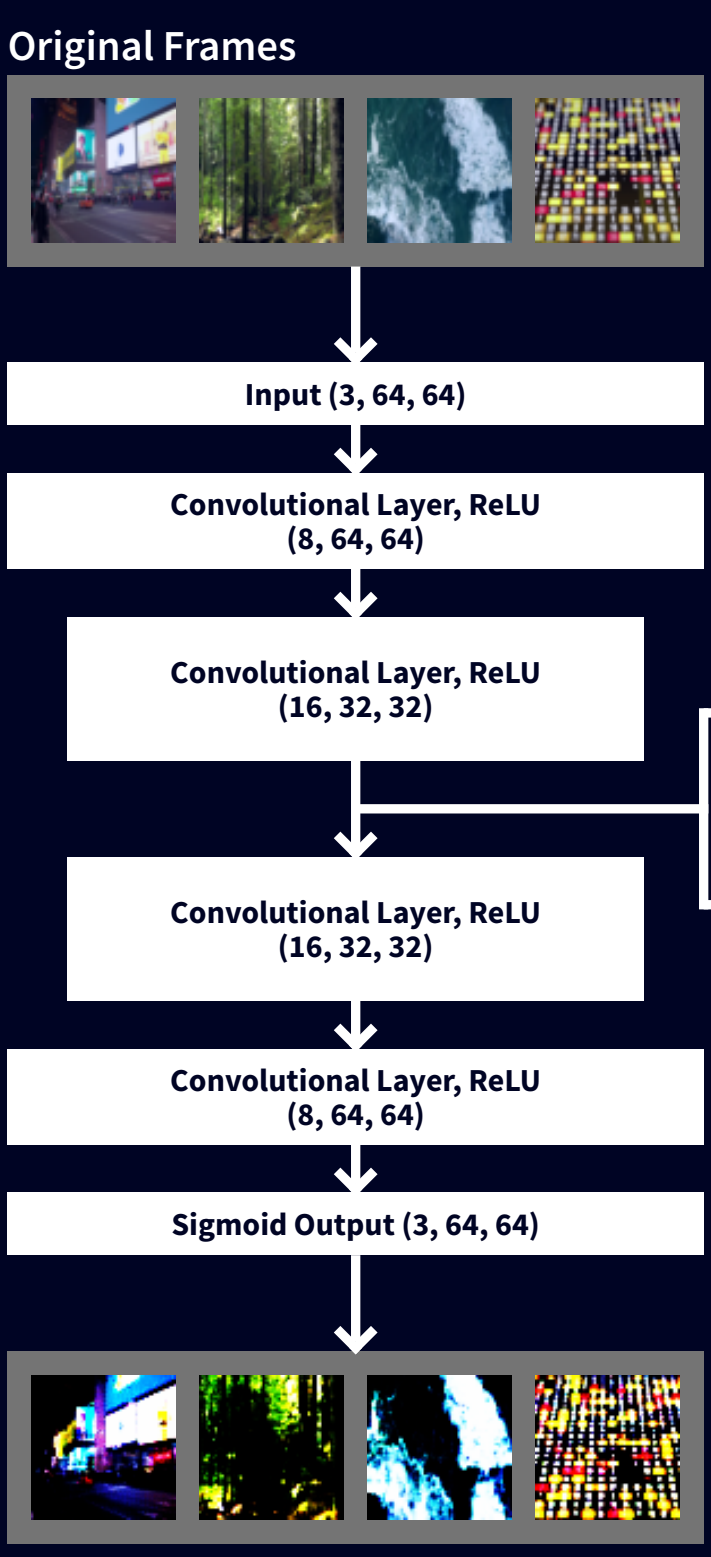
For Ocean, the video consists of a overhead shot of calm waves. The frames are mostly cyan, with whites in the frame where waves are moving.

For Stock, the video consists screen with the image on the screen panning down; this image consists of rectangles of bright light changing in color and sequence. The bright colors are red, orange and yellow.

Depicted to the right are four examples of the frame data used in the network which are the video frames fit into a 64 by 64 pixel grid. This data is has also been normalized using the mean and standard deviation from the frames in all 4 videos. The videos were shortened in time sequence to 10 seconds, and reduced in frame rate. Each video used for training retains the same number of 128 frames for the sequence processed by the network.

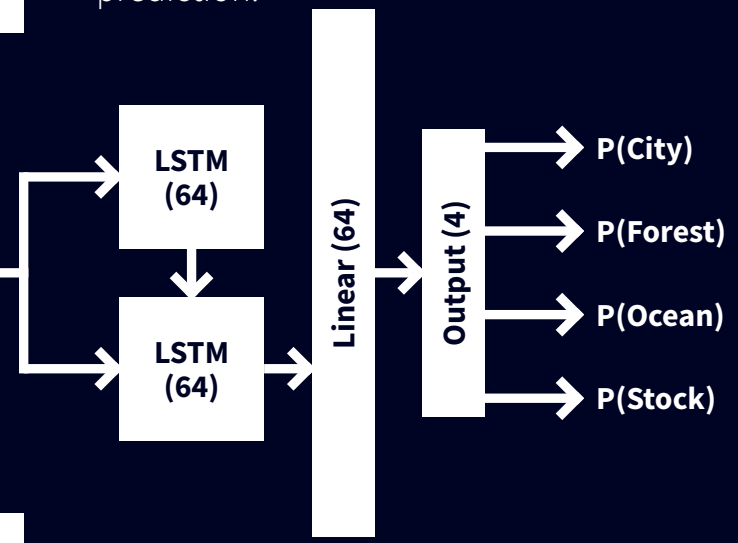
The 'surprise' videos consist of the beginning half of one video and the second half of another, with 64 frames dedicated to each. Besides the order, there is no difference in the data between the sequences input into the network.

Network Design



The auto-encoder network, which is depicted feeding from the top of the page to the bottom, was trained to reproduce the frames from the video in random order. This was done to create representations of the images that took advantage of CNN architecture.

The first half of this network, the encoder, was used as the input for the LSTM model. This allowed the network recognize sequences of reduced CNN representations for predicting video that was being shown. The network outputs probabilities for the prediction.

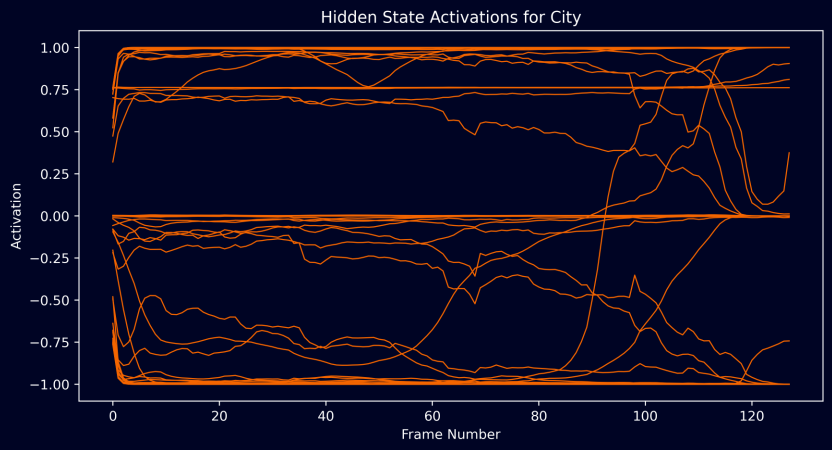


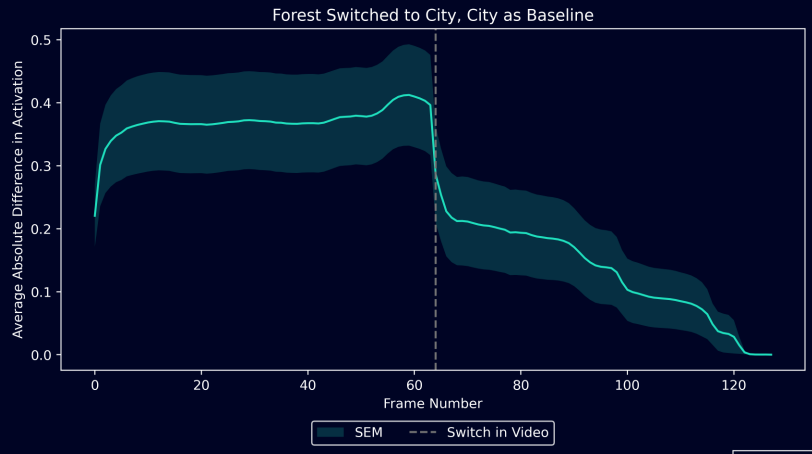
Both training sequences used frame data reduced to 64 by 64 pixels which were normalized with the mean and standard deviation of all data across all videos. The training used an initial learning rate of 0.007, which was programmed halve every 5 epochs where a reduction in Mean Squared Error (MSE) loss of less than 0.01% occurred.

The LSTM combined network was then fed videos frame by frame for both original videos and 'surprise' videos. The data was across frames was then used to create figures showing the difference within network over the course of video frames.

Activations: City

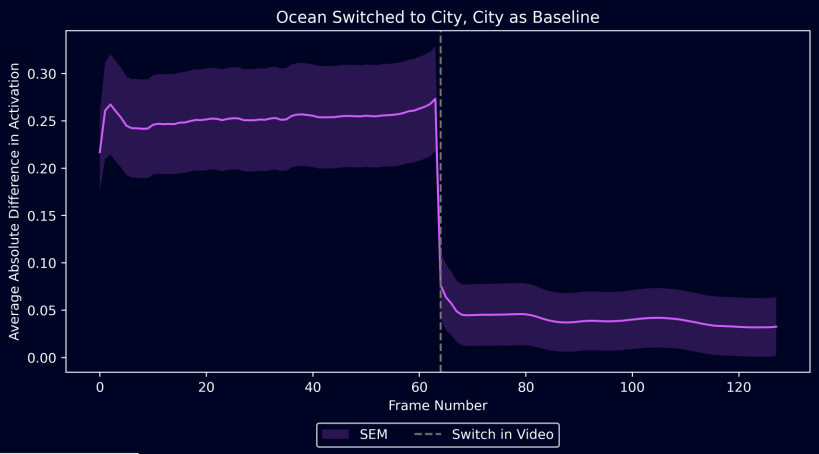
Depicted to the right is the 64 hidden activation states for the LSTM sequence sensitive module. The city video consists of mostly strong activations, in both the 1 and -1 boundaries of activation strength. There is some crossover from 0 to an activation extreme, or vice versa, but mainly towards the later frames in the sequence.

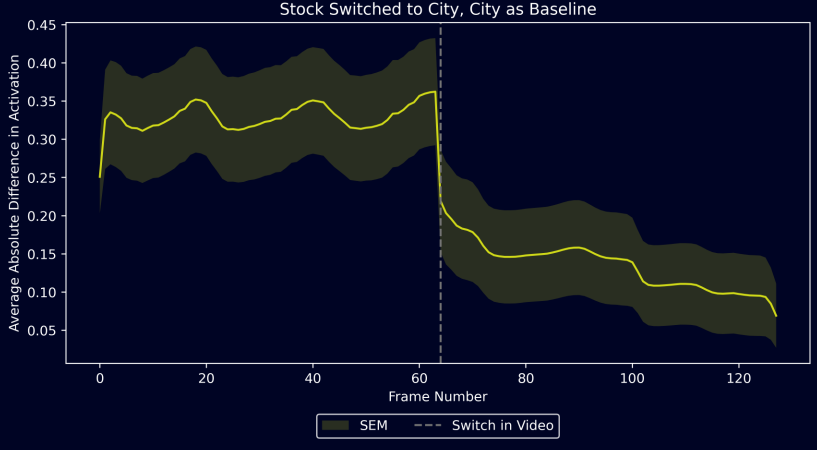




Depicted to the left is the average absolute difference in activation state value from the average activations in a video that starts Forest and ends City, with standard error of the mean (SEM) depicted in the shaded region. There is some rapid return to the City after the switch, with additional slow return to City activations over the given frames.

The switch from Ocean to City is a relatively drastic change decline in the average absolute difference, with the return being near complete. This suggests that the activations are highly differentiable and distinct for both Ocean and City.

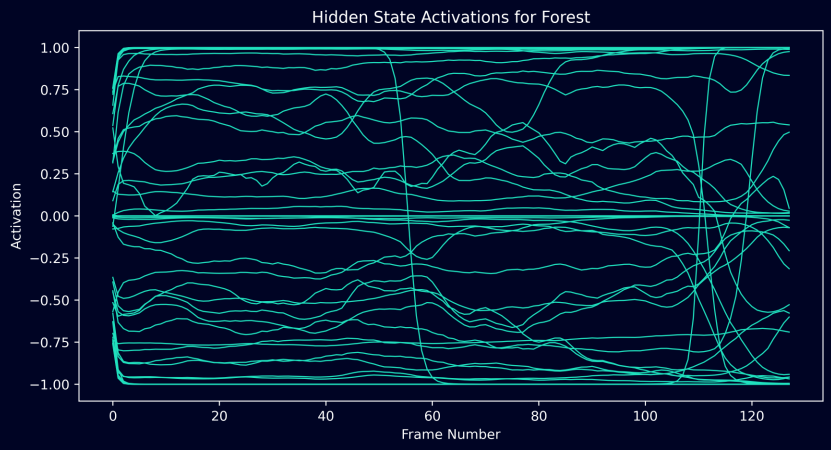


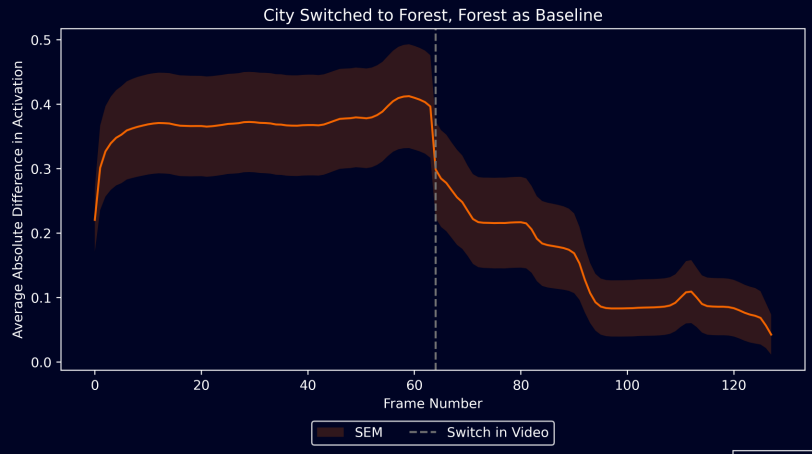


The switch from Stock to City is more similar to that from Forest to City. There is some return to the City after the video switch, with an additional slow return to City activation states for the given frames. The return to activations is less complete than Forest, likely due to similarity in flashing lights between the two videos.

Activations: Forest

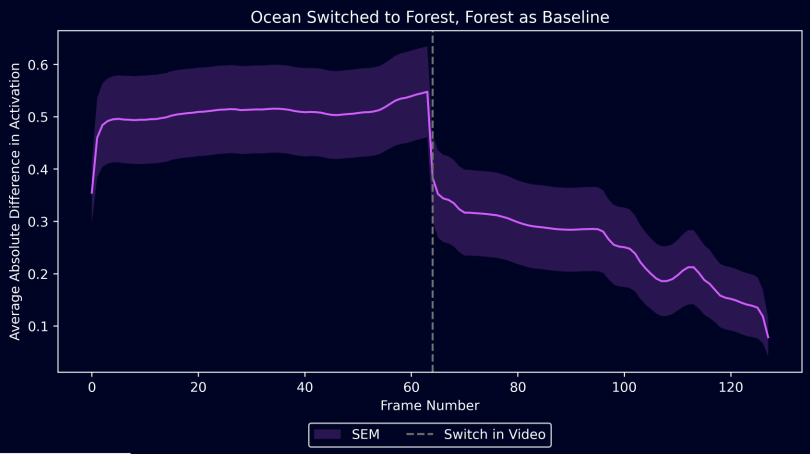
The hidden activation values for Forest are more evenly dispersed with wavy patterns that remain for most of the frame sequence. Another characteristic is the large changes in some of the sequence from one extreme value to the next in a relatively short course of time. This is a characteristic of Forest not seen in other video activation data.

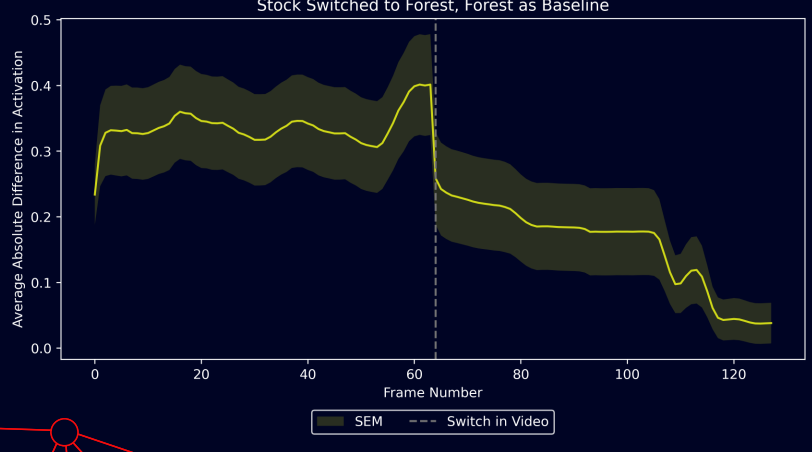




City switching to Forest shows large initial difference and a relatively small decrease in difference on the onset of Forest frames. This is seen in all of the videos for Forest as a baseline and characterizes the switch to Forest. Additionally here, there is a staggered return, which is most noticeable in the City and Stock videos.

The switch from Ocean to Forest is a small sharp decrease with a gentle return to normal Forest activations. This is likely due to Forest and Ocean being the more similar, being mostly of a single color and having slow movement.

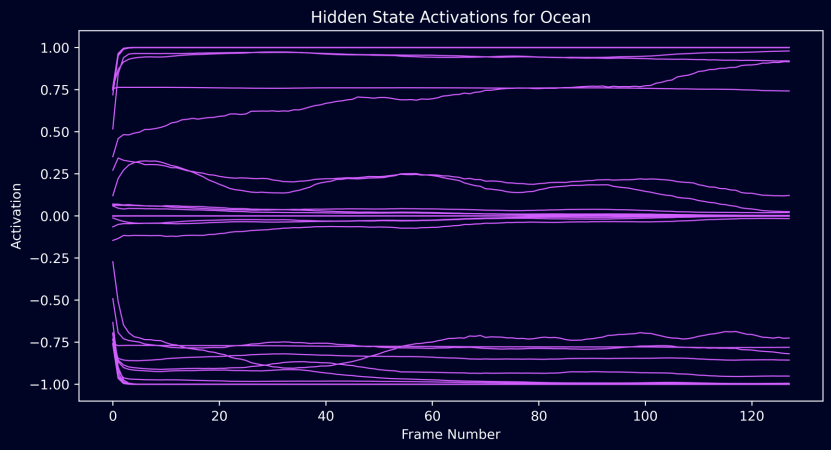


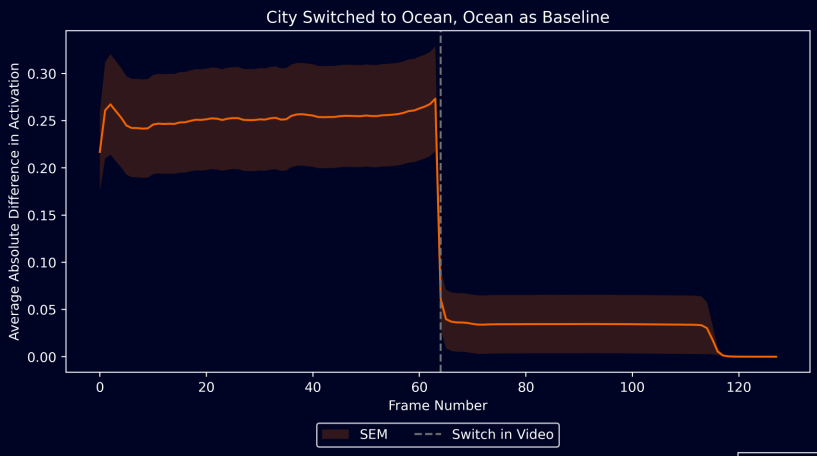


Stock generally mimics other activations after switch due to its oscillatory pattern covering the distribution of activations in its sequence. It shows here to have a gentle decline after initial change like Ocean with a greater change after sequence is established like in City.

Activations: Ocean

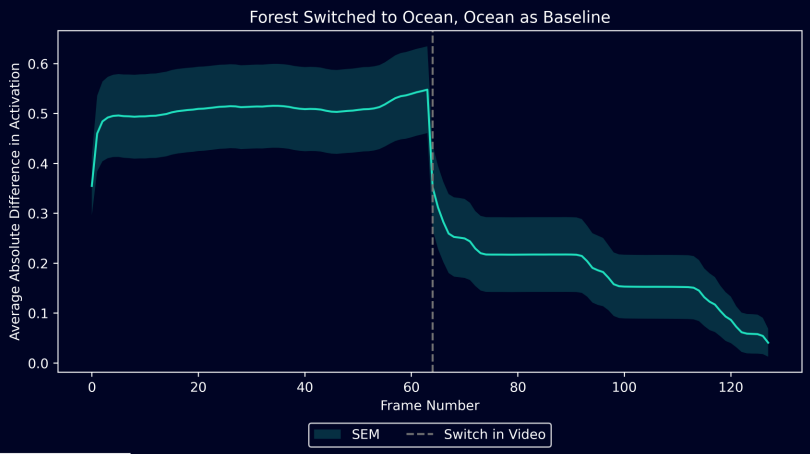
The activations in Ocean undergo the least change of any frame sequence. This is likely due to the small amount of change throughout the video with a relatively homogenous color palette and slow movement through the video frames. Activations for ocean converge quickly and remain in a similar level of activation for the entirety of the frame sequence.

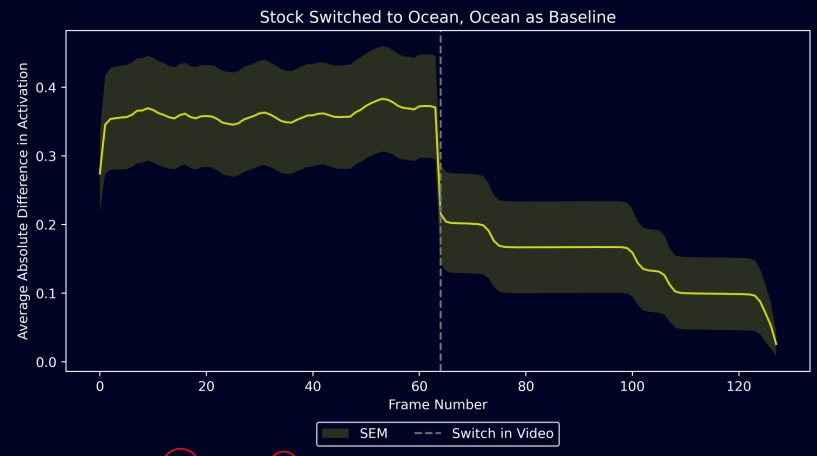




City switching to Ocean shows the largest initial difference among the group, with a small decrease later after sequence is better established. This again indicates that activations in Ocean and City are the most dissimilar between all of the videos and are the most dissimilar between a single frame between any of the videos.

The switch from Forest to Ocean is a small sharp decrease with a gentle return to normal Ocean activations. This is similar to the switch seen from Ocean to Forest and indicates some similarity in the activation states resulting from the two videos.

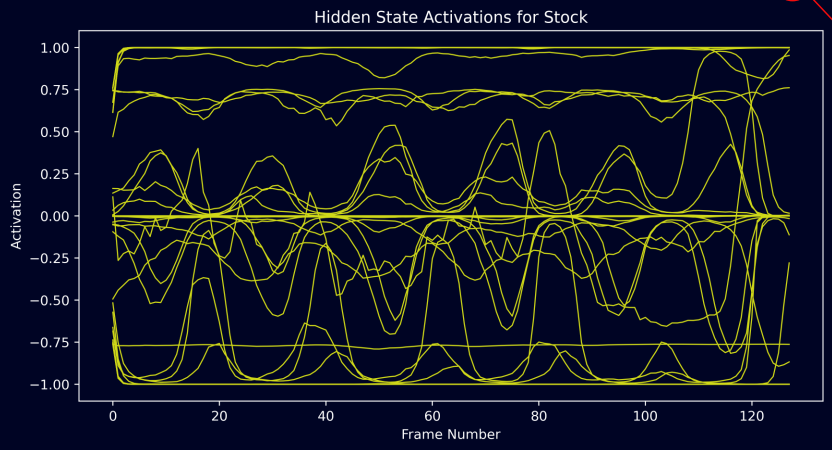


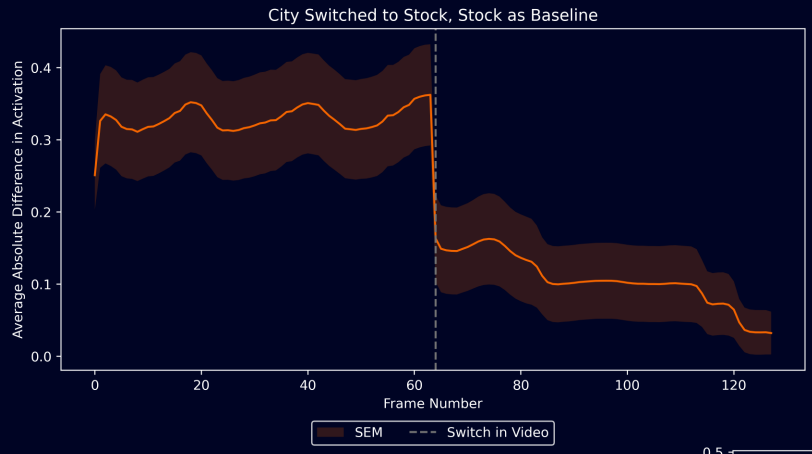


Again with stock there is a deceptively gentle decline in the return to activations for the baseline video. The characteristic oscillatory pattern can again be seen in both the initial difference and in the return of activations to Ocean.

Activations: Stock

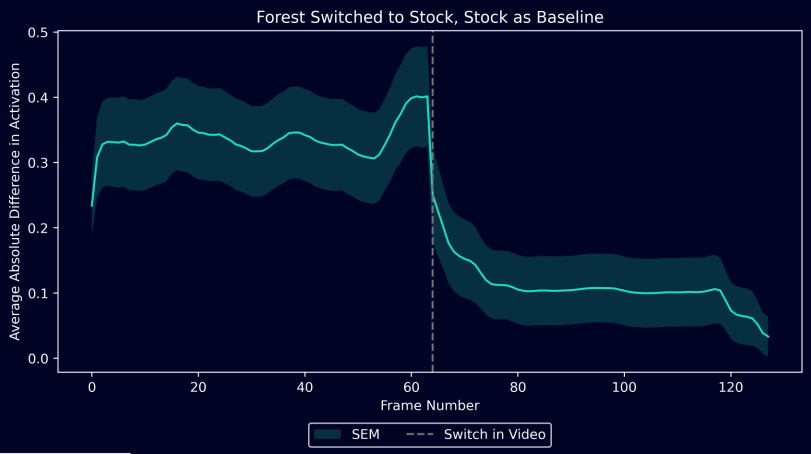
Activations from the Stock video frame sequence are distinct in their regular fluctuations from one extreme, to zero, then back again. This oscillation is the hallmark of Stock and makes the activations difficult to pinpoint due to their regular change that covers the entirety of the frame sequence. This pattern can be seen in the differences between videos as well.

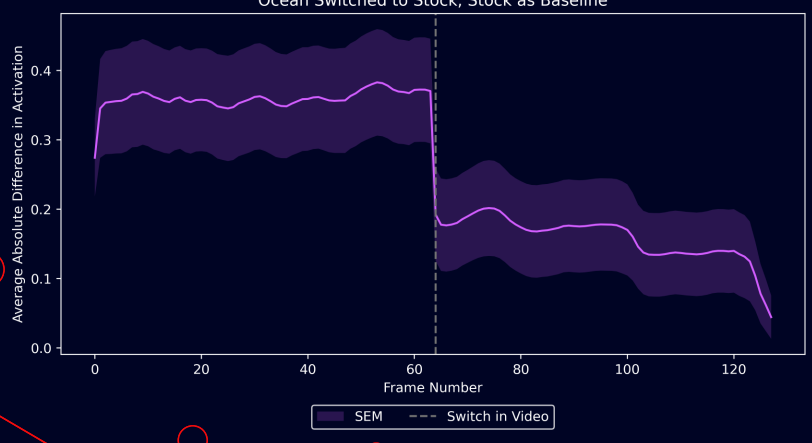




City switching to Stock shows the characteristic gentle return, which is present due to the Stock activations presenting at similar values to the other videos during the frame sequence. Also seen is the characteristic oscillatory pattern in the differences, which is seen in all videos where Stock is the baseline for comparison.

There is a gentle parabolic return here from Forest to Stock. This is likely due fluctuation patterns being present in both frame sequence activations from the network. It should be noted that the return is near complete, which is a characteristic of a return to stock activations.





Ocean switched to Stock shows the least immediate return to Stock activations after the switch. This is likely due to the Ocean values remaining relatively stagnant throughout the sequence, aiding in reduced difference over time to oscillatory activations brought on by the switch to Stock.

Probabilities: City

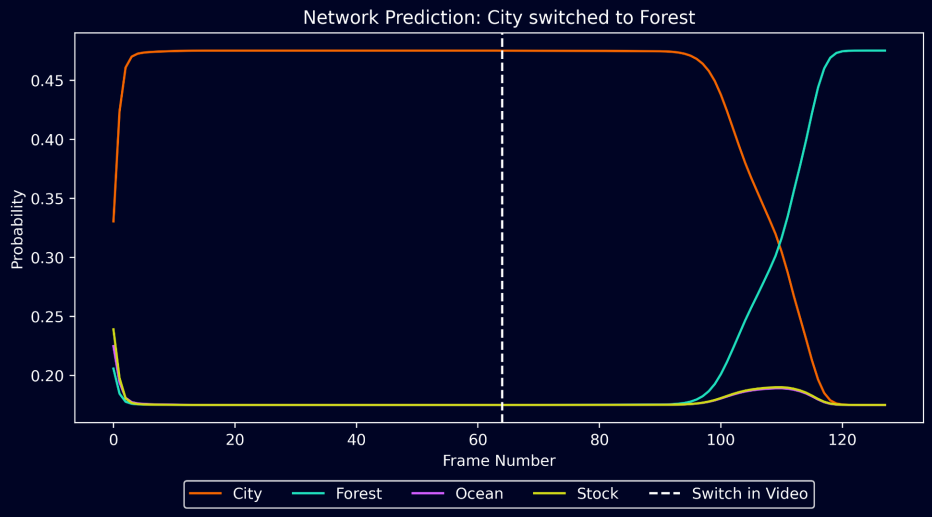
The outputs of the network over the course of the video sequences are depicted to the right. The decision the network makes for the type of video that is being presented is chosen based on the greatest of the 4 probability outputs.

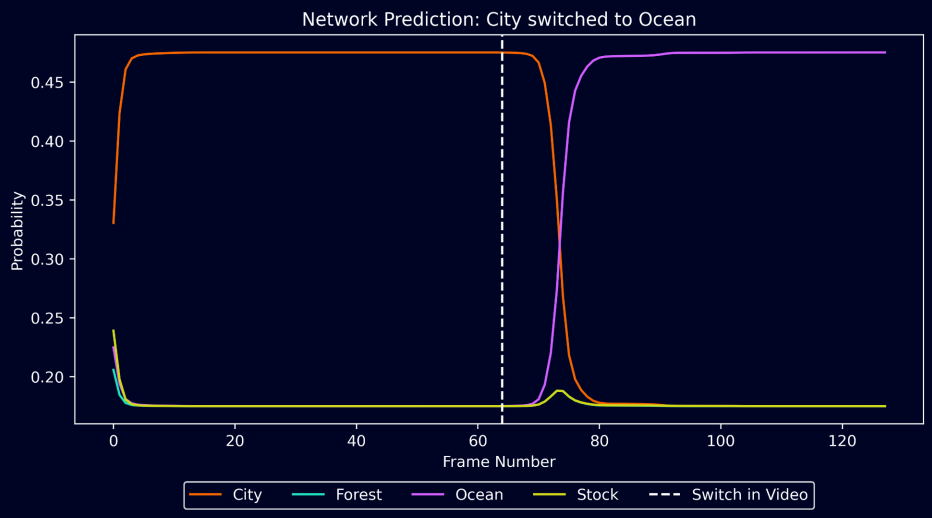
As seen for the City videos the maximum certainty of the network is decided within 10 frames presented to the network, with the network beginning with City as the highest probability and the choice of video type.

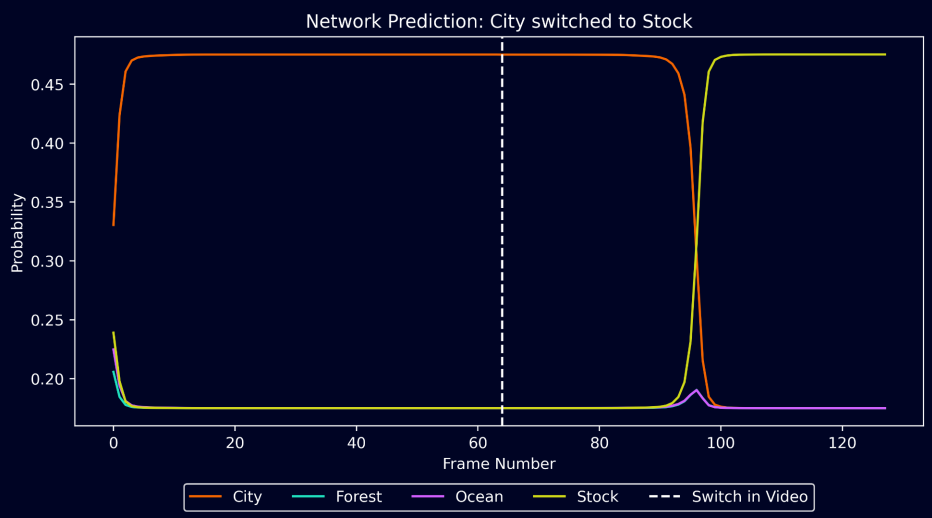
After the switch the probabilities for all videos remain for some time before switching to the highest probability being the one presented at the end of the frame sequence.

Switches to Ocean and Stock seem similar in that they are suddenly switched to the correct video after a certain number of frames under which the decision is made, with Stock taking more frames to make a decision because of its similarity to the City video.

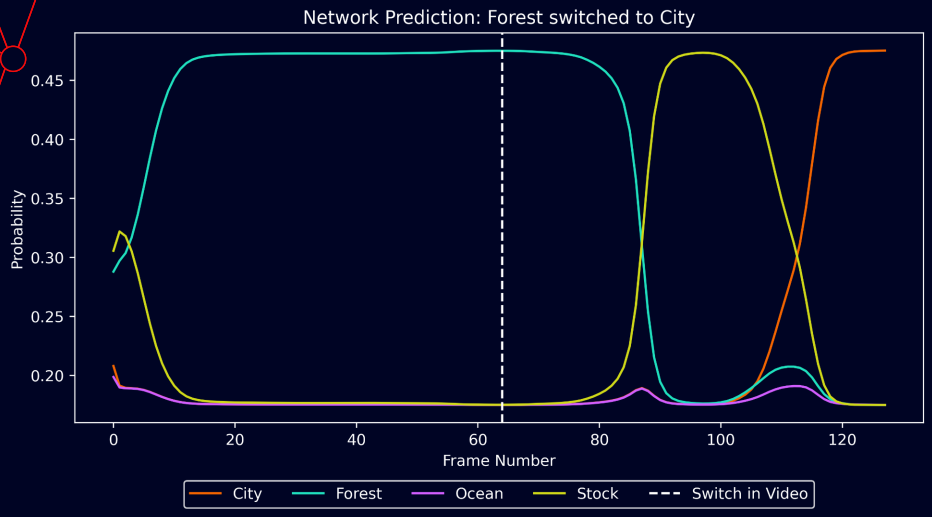
Forest takes the most time to and has a slower switch, indicating that the network is least immediately certain after the stimulus changes.

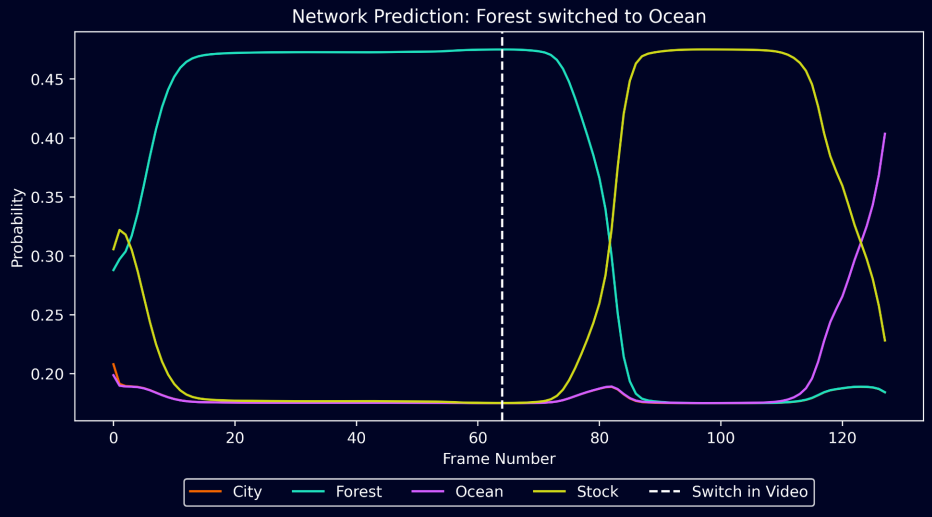


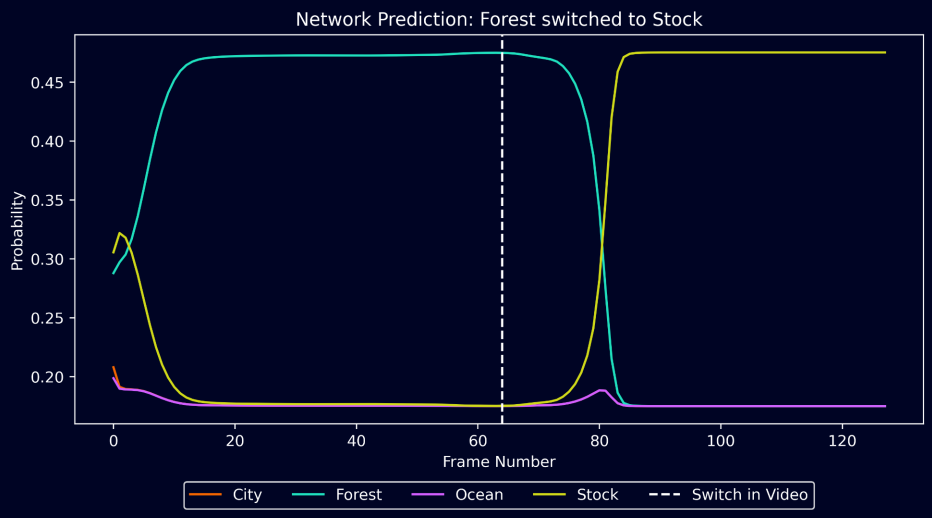




Probabilities: Forest







The videos that begin with Forest are characterized by a long sequence of wrong predictions after the switch, being dominated by Stock, except, of course, in the case of a switch to Stock.

The Switch to City and Ocean only happen after a major misclassification by the network. This is again due to the quirk in the Stock videos being oscillatory, as well the simple nature of the network which works on limited data representations.

City finally returns to the highest classification after over 45 frames, where it then reaches the peak probability for the network.

Ocean is another story. It only reaches correct classification with very few frames left in the sequence. Additionally it does not reach maximal probability for the network within this short frame sequence, though it likely would return shortly given a longer sequence.

As for Stock, there is a return to the Stock prediction in its entirety without misclassifying as another video entirely. Stock returns relatively quickly but is not the quickest return of any video, that being City returning to Ocean.

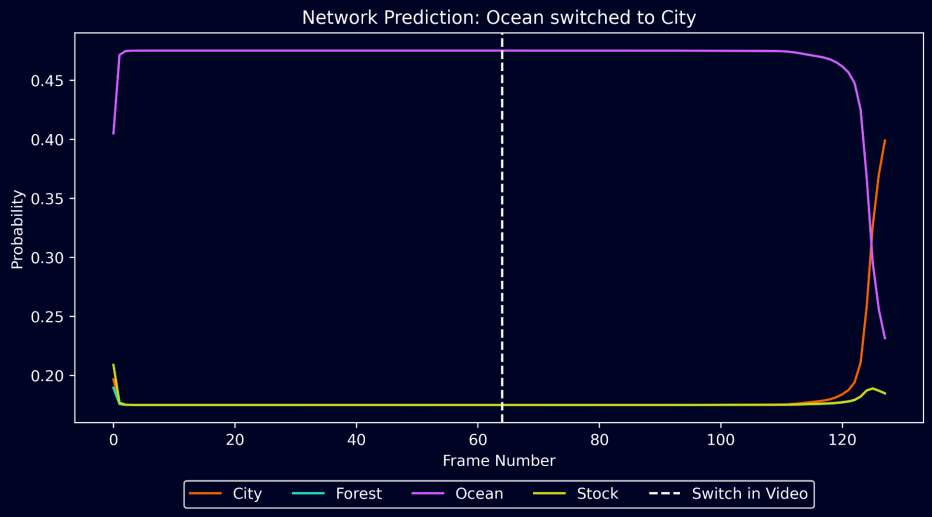
Probabilities: Ocean

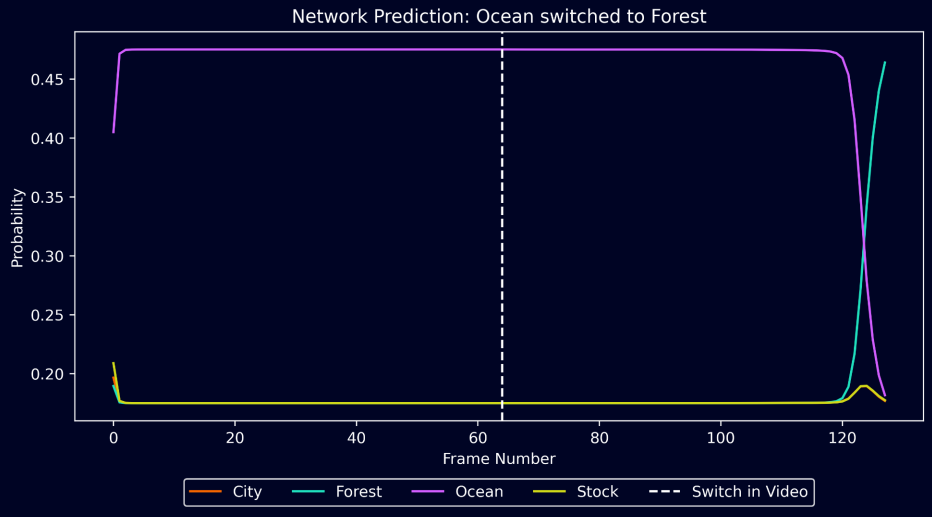
The probability output for videos beginning with Ocean are seemingly resistant to change. This is likely due to a strong recursion of repeated, slowly changing hidden state activation values.

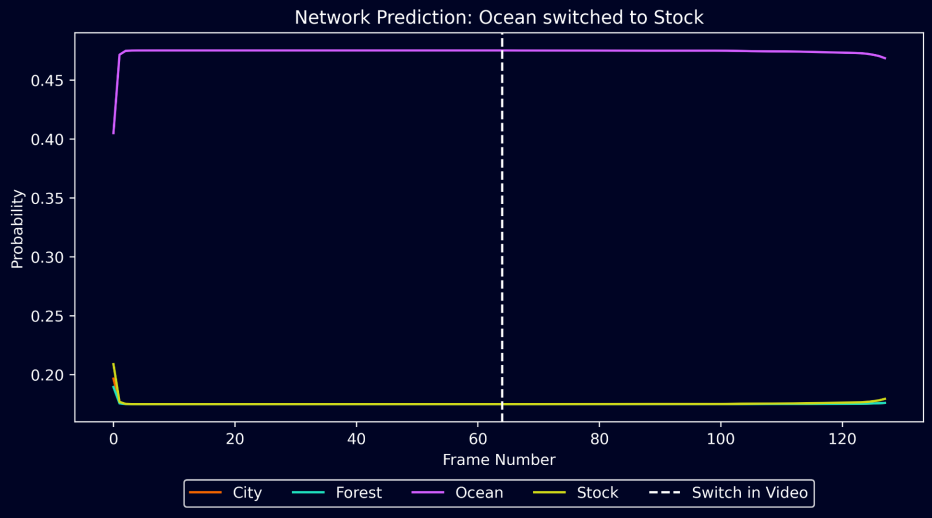
For the City and Forest endings, the switch from the incorrect to correct prediction does not happen until near the very end. Additionally the maximal probability for the correct prediction is never reached, with Ocean being the one that gets the closest. This may be due to Forest video adopting distinct activations for the end of the video, as discussed in the activations section.

Ocean switching to Stock never changes to the correct prediction of the second video during the course of the video sequence. It presents only a small downturn near the end of the sequence

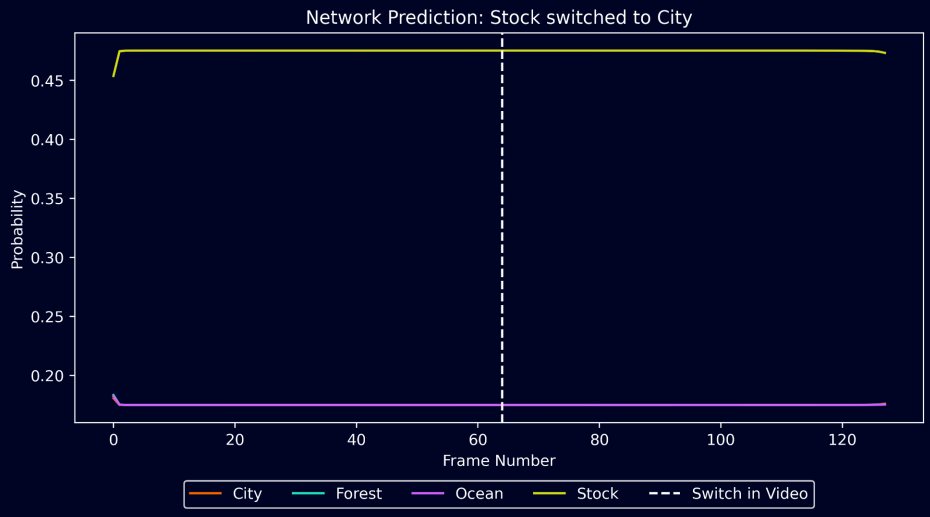
If the frame sequence were longer, it is likely that the downturn would lead to a switch in highest probability, especially considering the slight upturn in Stock seen at the same time.

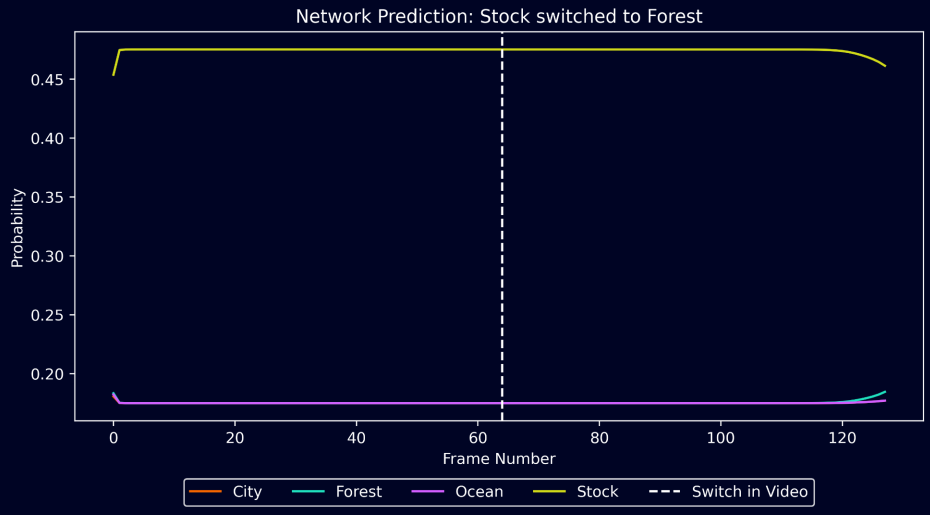


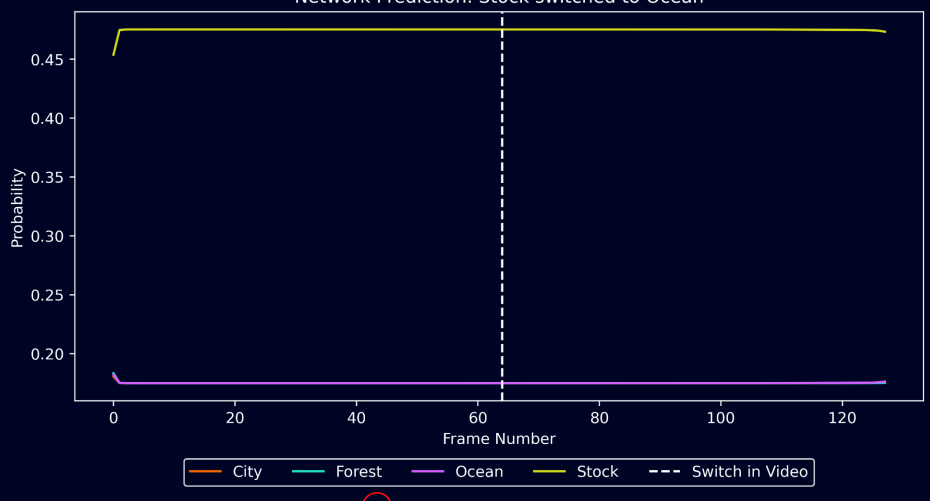




Probabilities: Stock







The Stock beginning videos are the most resistant to change after the video stimuli switches.

None of the correct classifications are found after the switch in the course of the video sequence.

Reasons for the resistance to change are likely those previously discussed in the previous sections, that being the oscillatory pattern that mimics activations at different time periods in the video, as well as the stable activations that mimic those that are found in other videos.

Slight downturns of the Stock probability can be seen near the ends of the frame sequences and slight upturns can also be seen at the same time for the correct classification probabilities.

This indicates that this is more resistance to change rather than a problem of misclassification as seen in the Forest probabilities.

Additional tests with longer sequences and more time periods within the videos would need to be done in order to fully assess the reasons for the resistance and when the switch in probabilities would occur.

Discussion

Through this proposal, data for the internal hidden activations of an LSTM model predicting the type of video can be seen. While reasoning was used in order to make sense of what was seen and explain why the differences occurred as they did, it is important to remember that this serves as a *proof of concept*. In order to properly asses reasons behind the shifts in activations or probabilities a greater depth of analysis would need to be undertaken, using statistical tests in order to assess differences not only between the averages of activations between video types, but also the activations themselves. A greater understanding of network dynamics cannot be determined conclusively without a greater knowledge of the both the network and the data presented.

The model used in this presentation was a highly simplified version of a more modern DNN. A greater complexity could be used for a subsequent model that may better represent the dynamics of data processing or yield representations more similar to those made in a natural brain brain. It cannot be known how well the network produces brain-similar representations without extensive testing on actual brain representation data, most likely utilizing Multivariate pattern analysis to compare dissimilarity between representations (Guggenmos et al. 2018). in light of this, components from reproduced networks already shown to retain comparative similarity to brain representations could be utilized, aiding in the justification of neural networking practices used.

An additional consideration the data here represents a low resolution and low frame rate representation of source material which depicts only a small amount of 4 different videos. Beyond the low dimension data, the sample size of 4 represents an obvious problem in the confines of traditional amchine learning for data classification. The Autoencoder is able to represent the images relatively similar to the input sequence (MSE loss of 0.5106 at 100 epochs) without cause for concern using unsupervised learning. The LSTM model would be considered 'overtrained' by the standards of neural network classification. Epoch 25 was used for being the first occurrence of the minimum loss for the training session over 100 epochs, however given there are only 4 data sequences the network is not suitable for actual classification in the modern sense. It speedily reached 100% accuracy by epoch 3 and it is likely this network, due to the simplicity and overtraining, would fail to properly classify or recognize sequence differences with any novel data, only being able to perform this *proof of concept* proposal due to the same image sequence being used for the entirety of the experiments, from training to analysis.

Never the less, this serves to show that there are interesting results in the investigation of surprise mechanisms in DNNs. This proposal serves to prompt further study into surprise and the use of DNNs in research.

Sources and Accreditation

–Citations–

Khaligh-Razavi, S. M., & Kriegeskorte, N. (2014). Deep supervised, but not unsupervised, models may explain IT cortical representation. PLoS computational biology, 10(11), e1003915.

Baker, N., Lu, H., Erlikhman, G., & Kellman, P. J. (2018). Deep convolutional networks do not classify based on global object shape. PLoS computational biology, 14(12), e1006613.

Zhuang, C., Yan, S., Nayebi, A., Schrimpf, M., Frank, M. C., DiCarlo, J. J., & Yamins, D. L. (2021). Unsupervised neural network models of the ventral visual stream. Proceedings of the National Academy of Sciences, 118(3), e2014196118.

Zhou, B., Khosla, A., Lapedriza, À., Oliva, A., & Torralba, A. (2015). Object Detectors Emerge in Deep Scene CNNs. CoRR, abs/1412.6856.

S. Ji, W. Xu, M. Yang and K. Yu, "3D Convolutional Neural Networks for Human Action Recognition," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 35, no. 1, pp. 221-231, Jan. 2013, doi: 10.1109/TPAMI.2012.59.

Guggenmos, Matthias, Philipp Sterzer, and Radoslaw Martin Cichy. “Multivariate Pattern Analysis for MEG: A Comparison of Dissimilarity Measures.” NeuroImage 173 (2018): 434–47. https://doi.org/10.1016/j.neuroimage.2018.02.044.

–Accreditation–

ChatGPT 4 was used in the course of development for this project.

Special thanks to Susie Ju for aiding in the design of this proposal.