Personal Project Summary

By Alec Knobloch

Problem Statement

Purpose:

I don't want to listen to (e.g.) Cialis advertisements when I watch Star Trek with my dad. This device will prevent commercials from intruding on the attention of Television watchers by muting potentially offensive/intrusive/repetitive/unwanted commercials.

Concept:

A program that mutes the audio on a TV or Computer in response to a commercial coming on screen in real time.

Approach/Steps

Implementation:

An in-line Device running a Program receives a copy of visual data sent to the TV screen and samples/tests every few frames/seconds to analyze and determine if a commercial is playing. Upon detection, the device emits an infrared signal that is identical to the signal the remote control would use to mute the television.

Reasoning:

This solution would require the least invasive additions to existing TV equipment as well as give development flexibility to work with a variety of different entertainment setups. In future versions I hope to have an even smaller footprint and be more integrated with existing equipment.

Method/Discussion:

Using the python cv2 class I can extract different features from each frame like Brightness, Sharpness, Contrast, and Visual Noise.

Audio signals at this time are not being considered as the muting a TV program would interfere with them.

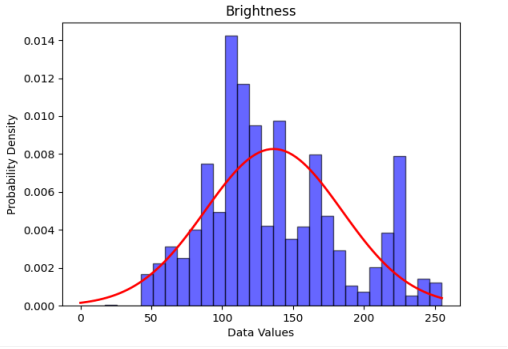
I weighed various classification algorithms, and chose to create a Gaussian Multivariate Distribution model that can take multiple features as input and output a probability in an efficient manner. I decided to develop two models, one trained on TV programming, and the second trained on Commercial programming. For each frame of a show the program distills values for each feature and inputs them into each model. Each model will output a probability that represents the likelihood that the tested frame is of its trained category. The two probabilities are then compared, and the highest probability is taken as the true prediction.

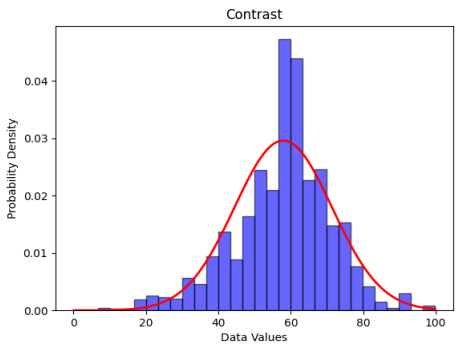
Results/Proof of Concept Testing

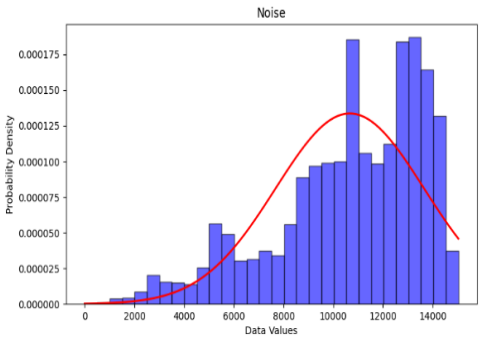
Testing:

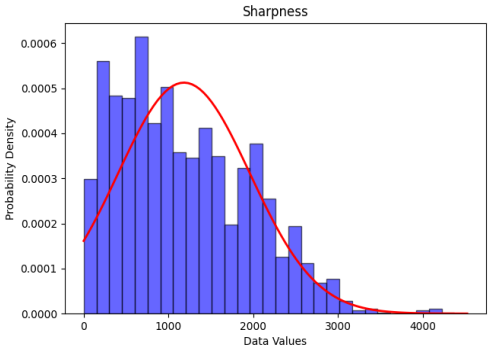
Feature Values of Star Trek Next Generation commercial periods

Commercial Frames:	Contrast	Sharpness	Noise	Brightness
Mean	57.91	1183.66	10664.67	136.21
Median	59.52	1039.22	10664.67	136.21
Standard Deviation	13.46	778.61	2985.51	48.24
Range	99.79	4525.85	15029.15	254.85
25% Quantile	50.89	552.25	9045.64	105.11
75% Quantile	66.28	1779.06	13080.38	166.25



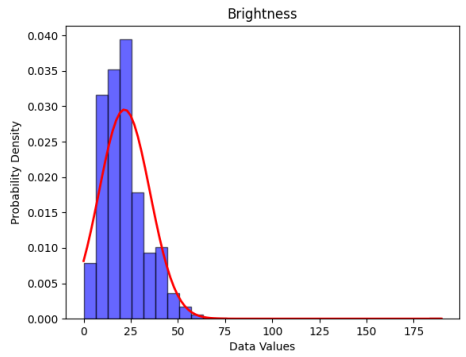


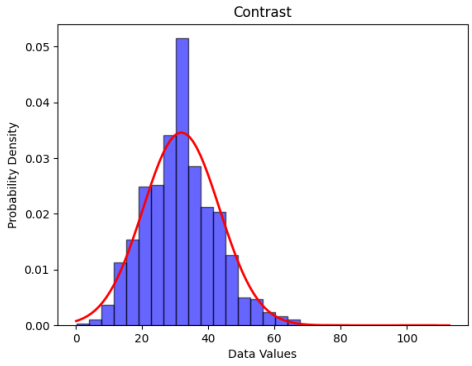


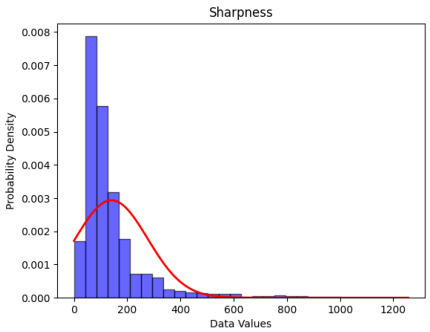


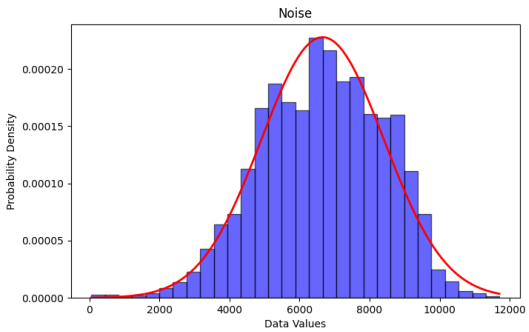
Feature Values of Star Trek Next Generation TV periods

TV Content:	Contrast	Sharpness	Noise	Brightness
Mean	31.83	140.88	6660.16	21.66
Median	31.47	101.13	6692.10	19.37
Standard Deviation	11.52	135.87	1751.92	13.50
Range	112.74	1255.55	11652.55	189.72
25% Quantile	24.122	67.04	5380.35	12.66
75% Quantile	38.30	153.54	7991.21	27.13









I made four models, one trained on Next Gen Star Trek, one on all the commercials during an episode of Star Trek Next Gen, one on the original Star Trek, and one trained on all the commercials during an episode of the original Star Trek. Here are the results of testing those models against each other.

TV and Commercial models	Results
Both models tested on Star Trek Next Generation Commercials	percent right: 98.40% wrong count: 361 --- 1.0112359523773193 seconds ---
Both models tested on Star Trek Next Generation TV Program	percent right: 98.04% wrong count: 1253 --- 2.8120551109313965 seconds ---
Both models tested on original Star Trek Commercials	percent right: 97.98% wrong count: 242 --- 0.5310404300689697 seconds ---
Both models tested on original Star Trek TV Program	percent right: 84.82% wrong count: 13183 --- 3.8705875873565674 seconds ---

The array of values is a cluster analysis of how many frames in a row one of the models will fail.

Average wrong cluster size original Star Trek model: 67.18888888888888
wrong cluster median size: 13.5
[3, 1, 79, 83, 5, 2, 84, 8, 77, 65, 32, 76, 29, 57, 67, 1, 1, 12, 47, 18, 5, 17, 3, 9, 1, 23, 6, 4, 2, 1, 3, 1, 3599, 7, 1, 50, 1, 1, 3, 57, 1, 1, 1, 5, 2, 2, 7, 35, 38, 3, 81, 1, 9, 36, 61, 39, 20, 25, 23, 7, 8, 1, 10, 15, 3, 31, 78, 17, 10, 17, 5, 1, 282, 1, 72, 46, 17, 50, 22, 52, 1, 44, 50, 51, 54, 7, 29, 72, 60, 3]

The Cluster of ‘3599’ incorrectly predicted frames is a commercial that was not manually identified and is actually the model working as designed.

Testing Conclusion:

The AdMute project successfully demonstrates a real-time solution for muting commercials by analyzing video frames. By leveraging computer vision techniques and a Gaussian Multivariate Distribution model, the system effectively distinguishes between TV programming and commercials based on extracted visual features such as brightness, sharpness, contrast, and noise.

Future improvements could involve optimizing classification accuracy, program efficiency, and refining the device’s footprint for broader compatibility with different entertainment setups.

Potential Next Steps

Check for motion of picture (high motion = likely Commercial AND 0 motion = likely Commercial):

Check for speed of cuts (High cuts per min = likely Commercial):

Check for likelihood of commercial based off time since last Commercial (Higher amount for time since last Commercial = more likely Commercial)

Check for amount of green in picture (NFL mode) (Higher Green in frame = unlikely Commercial)

Create a short window that of memory in the model to smooth out the average cluster of fails.)

Appendix/Working Notes:

AdMute Project

https://github.com/Knobloch-design/Commerical-Analysis