Context-aware encoding and dynamic encoding ladders

Machine Learning for Per -Title Encoding

- Navid Shahbazi
- Nasim Jamshidi Avanaki
- Mohammad Al-Diabat

Outline

- ❑ Resolution Switching and convex hull
- ❑ Subjective and Objective Quality Assessment
- ❑ Problem
- ❑ Dataset and data preprocessing
- ❑ Training the model
- ❑ Results and discussion

One-Size-Fits-All encoding

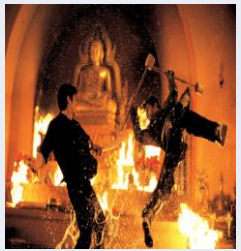
- **Motion** ⟶ **Low vs High**
- **Texture** ⟶ **Plain vs Noisy (Complex)**

- **Downside**
- For the scenes with high complexity ⟶ Blockiness
- For simple content like cartoons ⟶ Waste the bitrate

Bitrate (kbps)	Resolution
235	320x240
375	384x288
560	512x384
750	512x384
1050	640x480
1750	720x480
2350	1280x720
3000	1280x720
4300	1920x1080
5800	1920x1080

bitrate ladder

Low Complexity



High Complexity

Subjective Quality Assessment

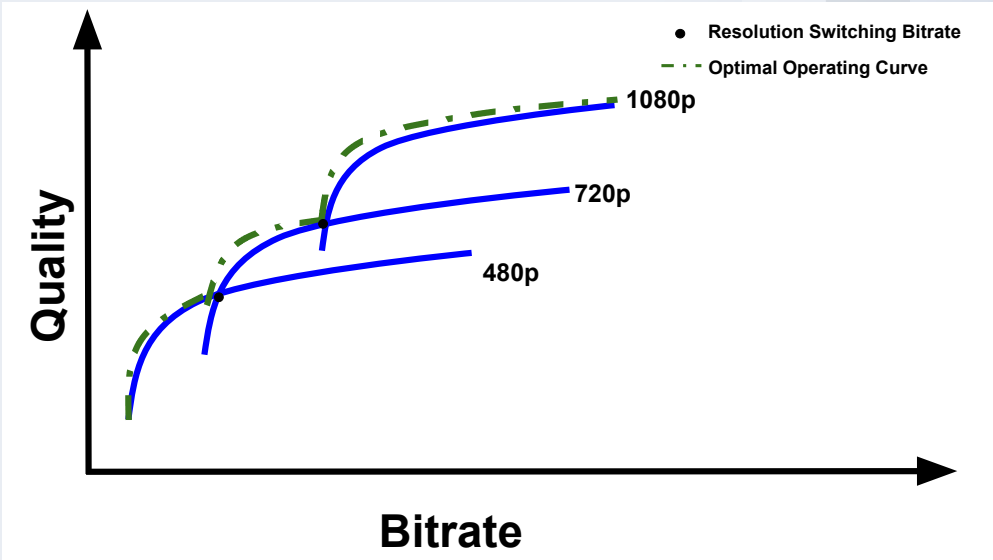
- **Typically we use subjective rating to find out the resolution switching points**
  - Subjective tests are expensive
  - Subjective ratings are not always available
- **Objective metrics can be an alternative**
  - Cons: Adding the errors due to prediction errors
  - Pros: it can be applied simply to every encoded videos

Objective Quality Assessment

- **Signal based models**
  - Full reference, reduced reference, no reference
  - **Cons**: the video has to be encoded first
- **Parametric based models**
  - Based on the network and **encoding parameters**
  - **Cons:** Less accuracy as content information is not available
  - **Pros**: no need to encode the video
- **Hybrid models**

Resolution Switching

- **At a certain resolution, bitrate can increase the quality to a certain point, after that point the quality gets saturated**
  - We need to increase the resolution

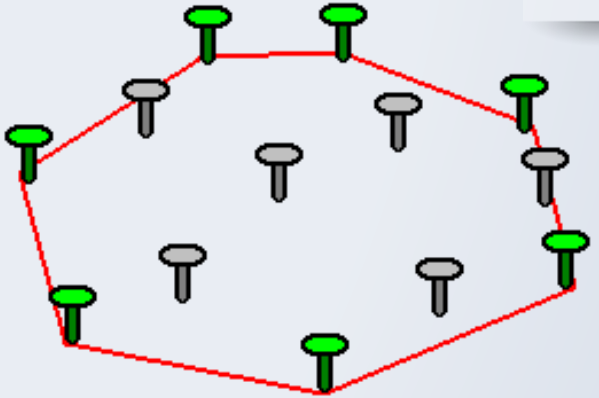


Resolution switching using objective metrics

- **Typically full-reference metrics are used**
  - **PSNR** (Peak Signal-To-Noise Ratio)
    - The most commonly used metric in video compression.
  - **VMAF** (Video Multi-Method Assessment Fusion)
    - Perceptual quality metric developed by Netflix
  - **Still expensive** procedure as the videos needed to be encoded in several bit-resolution pairs to get the curves
    - Can be replaced by parametric models

What is the Convex Hull ?

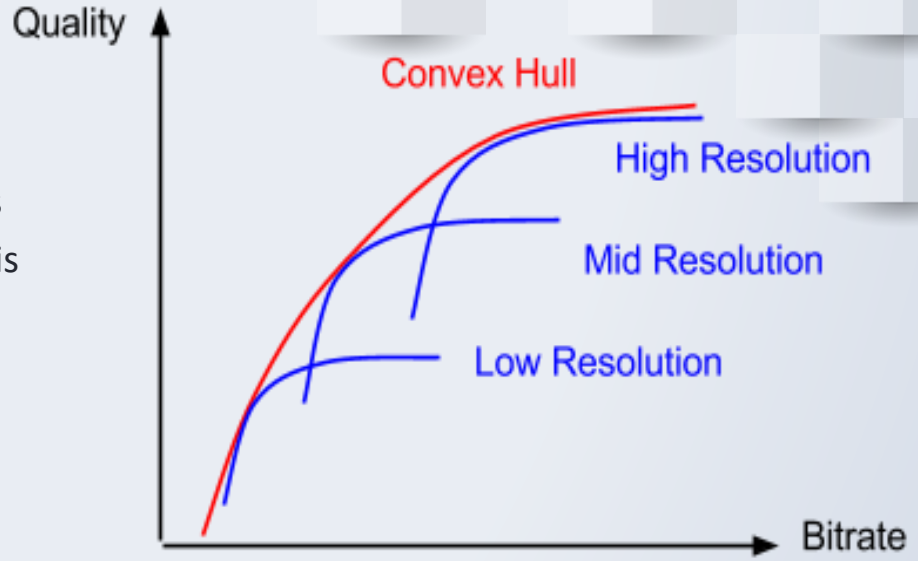
Rubber band



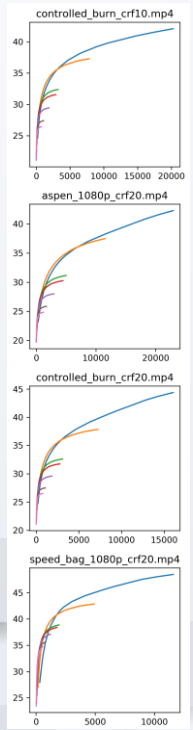
Convex Hull

What is the Convex Hull ?

- Mathematically, the convex hull or convex envelope or convex closure of a set X of points in the Euclidean plane or in a Euclidean space is the smallest convex set that contains X.

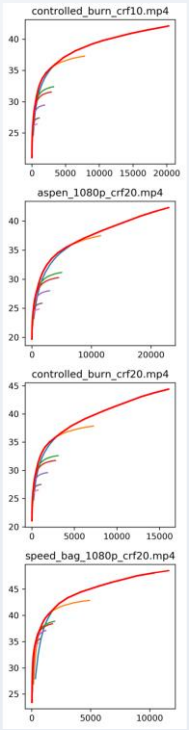


Convex Hull & Complexity Analysis (For each Movie/Clip)

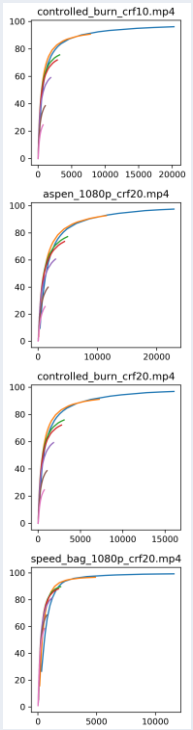


RD Curves

PSNR

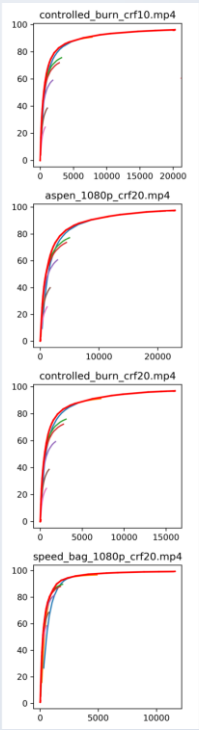


Convex Hull



RD Curves

VMAF



Convex Hull

Problem

- **Predict VMAF based on the encoding parameters as well as video information**
  - Pros: There will no need for encoding the video anymore
  - Challenge: the prediction strongly affected by content complexity
    - Temporal and spatial complexity

Dataset tables

- **Clip**: attributes of a clip.
- **Encode**: describes a single test encode with used encoding parameters (e.g. crf, resolution, etc) and derived quality metrics (VMAF, PSNR) of an encode.
- **Scene Change**: Scene changes indicate a change of the setting/scene in a video.
  - complex content has a lot scene changes
- **Label**: Labels for each source video which are tagged using machine learning classifiers like Tensorflow Mobilenet to define labels and categories for the content.

Data preprocessing

- **Missing values**
  - Nan values are removed
- **Outliers are removed**
  - We used ITU-T recommendation p.1401
    - It is designed for subjective rating, but found it to be useful
- **Categorical data handling**
  - Dummy Coding
- **Normalizing the data**
  - It is required as the features have different ranges

Predicting VMAF
[Clip_Encode] table

- **Predict VMAF purely based on the encoding parameters**
  - **Cons**: missing information about the video complexity
    - Assume that videos have similar complexity level
- **Features used for training**
  - **Clip table:**
    - clip_duration, clip_frame_rate, clip_height, clip_size
  - **Encode table:**
    - encode_WidthHeight, encode_bitrate_video, encode_crf

Predicting VMAF
[Clib_Encode_scenechange] table

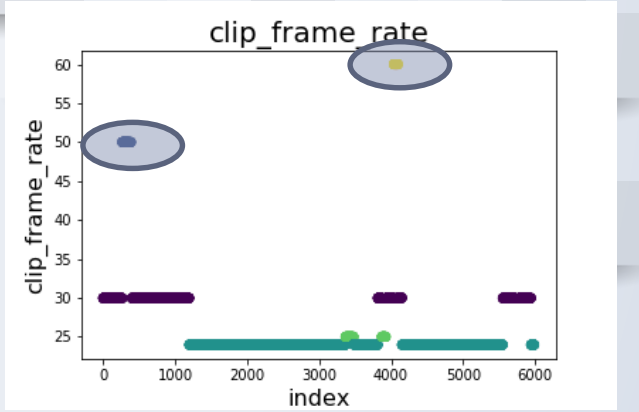
- **Predict VMAF based on the encoding parameters and scene change percentage**
  - **Pros:** taking into account the temporal video complexity
  - **Cons**: missing information about the spatial video complexity
    - Assume that videos have similar spatial complexity (texture) level
- **Features used for training**
  - **Clip table:**
    - clip_duration, clip_frame_rate, clip_height, clip_size
  - **Encode table:**
    - encode_WidthHeight, encode_bitrate_video, encode_crf
  - **Scene change table**: percentage of scene change in 1 second

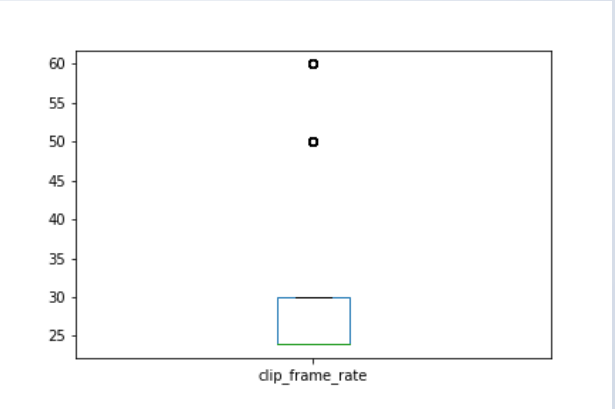
Training data

- **Train set and test set**
  - **Blind cross validation**
    - Really good results, but probably biased to the training set
  - **Split dataset**: 25% of source videos (based on clip table) are chosen to be test and 75% training set
  - **One-hold-out cross validation**: every time one source video was out trained based on the other sequences, then tested based on the holdout sequence
- **Three regressions are used:**
  - Linear regression, SVR (rbf kernel), random forest

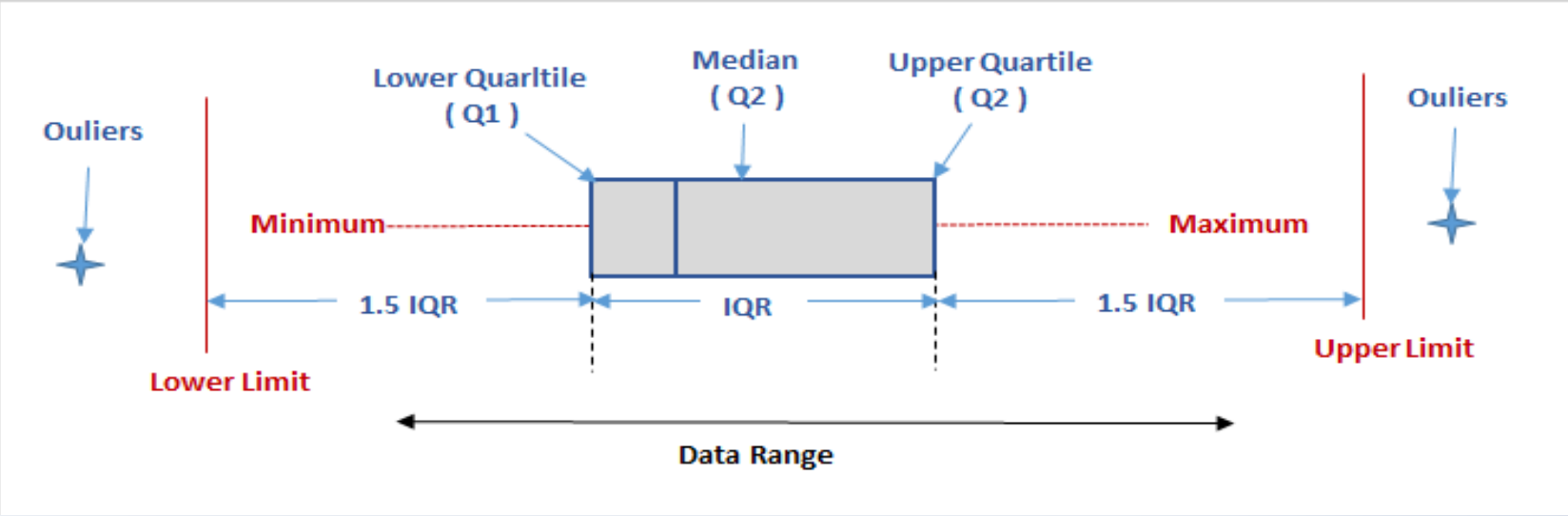
Data preprocessing

- **Data preparation**
  - **e.g. Framerate:** Change from 30000/1001 to 30 fps
  - Nan values are removed
- **Outliers are removed**
  - Check them in simple scatter plot first
  - Box-and-whisker-plot as similar to ITU-T p.1401 the outlier are removed





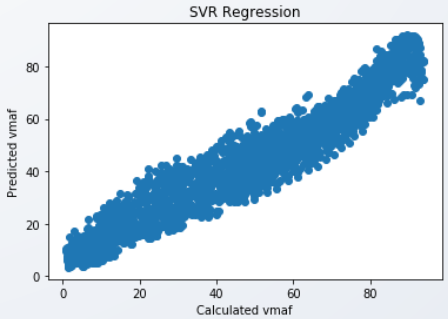
Outlier detection: Box-and-whisker-plot

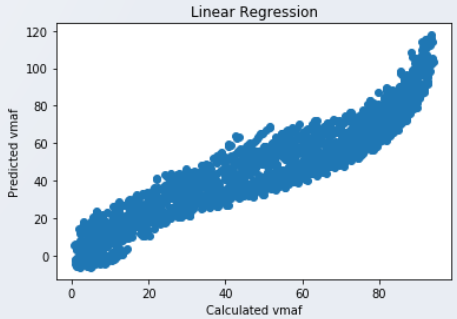


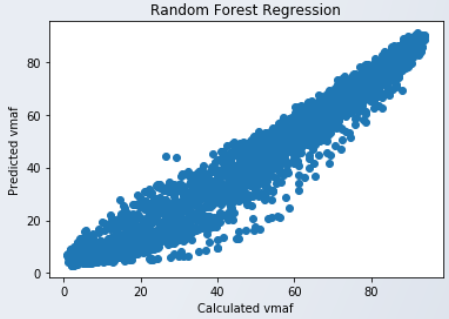
Plot from Quality and Usability Seminar

Training and results

- Split dataset [Clip_Encode tables]



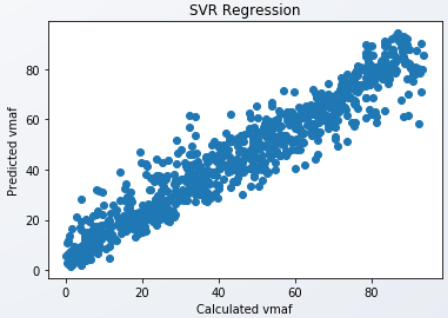


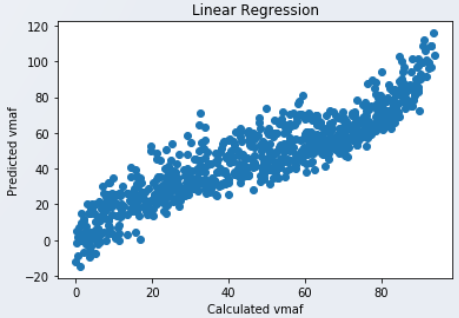


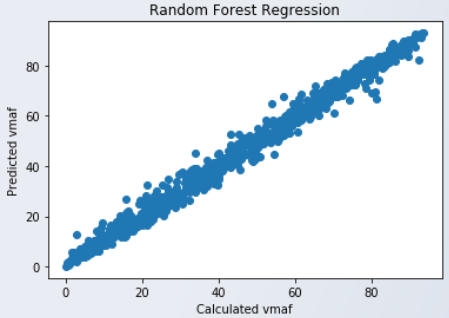
	SVR	Linear	Random Forest
MSE	86.1827	101.0071	86.9497
RMSE	9.2834	10.0502	9.3246
Pearson Correlation	0.9588	0.9344	0.9654

Training and results

- **Blind cross validation [Clip_Encode_Scene change tables]**



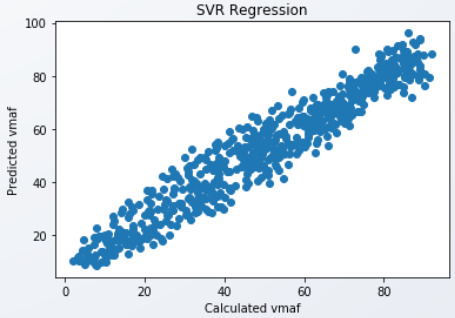


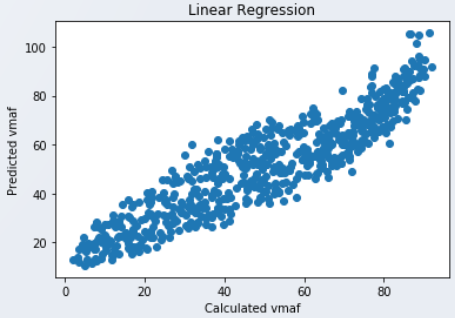


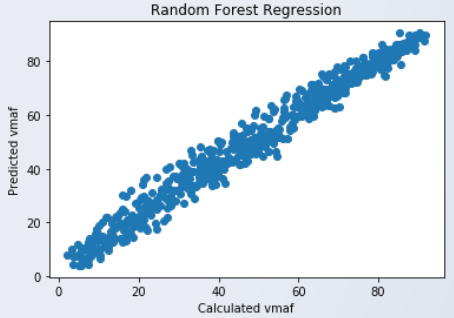
	SVR	Linear	Random Forest
MSE	66.4030	119.8154	8.8470
RMSE	8.1488	10.9460	2.9743
Pearson Correlation	0.9534	0.9095	0.9936

Training and results

- Split dataset [Clip_Encode_Scene change tables]







	SVR	Linear	Random Forest
MSE	48.6618	99.7779	19.8211
RMSE	6.9758	9.9888	4.4520
Pearson Correlation	0.9647	0.9146	0.9857

Next Steps

- One-hold-out cross validation
- Improving the results by adding content labels (Mobilenet labels)
- Importance of features for training
  - Based on XGBoost
  - Based on Information Gain
- Predicting Convex hull using our trained model

THANKS!

Any questions?