nofu – A Lightweight No-Reference Pixel Based Video Quality Model for Gaming Content.

Steve Göring, Rakesh Rao Ramachandra Rao, Alexander Raake;
Audiovisual Technology Group, Technische Universität Ilmenau, Germany;
Email: [steve.goering, rakesh- rao. ramachandra-rao, alexander.raake]@tu-ilmenau.de

based on QoMEX 2019 paper: https://bit.ly/31i0jcZ

October 16, 2019
Motivation – Gaming Streams

- beside classical video streams → gaming content:
  - e.g. Youtube Gaming, Twitch, ...

- gaming videos →
  - additional requirements / properties: Zadtootaghaj et al. [9]
    - live streaming, low delay, low stalling,
    - high video quality, cgi content, streaming technology

- focus on video quality of gaming streams
  → gaming qoe and gaming video quality
Gaming QoE and video quality

- several influencing factors: Möller, Schmidt, and Zadtootaghaj [8]
  - video quality factors: content (cgi), encoding (fast),
  - interaction: delay, ...

- objective full-reference metrics: good results: Barman et al. [1, 2, 3]
  - VMAF best; problem: reference usually not available

- for live/adaptive encoding:
  - fast, accurate, no-reference quality estimation

→ nofu
Gaming QoE and video quality

- several influencing factors: Möller, Schmidt, and Zadtootaghaj [8]
  - video quality factors: content (cgi), encoding (fast),
  - interaction: delay, ...

- objective full-reference metrics: good results: Barman et al. [1, 2, 3]
  - VMAF best; problem: reference usually not available

- for live/adaptive encoding:
  - fast, accurate, no-reference quality estimation

→ nofu
Gaming QoE and video quality

- several influencing factors: Möller, Schmidt, and Zadtootaghaj [8]
  - video quality factors: content (cgi), encoding (fast),
  - interaction: delay, ...

- objective full-reference metrics: good results: Barman et al. [1, 2, 3]
  - VMAF best; problem: reference usually not available

- for live/adaptive encoding:
  - fast, accurate, no-reference quality estimation

→ nofu
Gaming QoE and video quality

- several influencing factors: Möller, Schmidt, and Zadtootaghaj [8]
  - video quality factors: content (cgi), encoding (fast),
  - interaction: delay, ...

- objective full-reference metrics: good results: Barman et al. [1, 2, 3]
  - VMAF best; problem: reference usually not available

- for live/adaptive encoding:
  - fast, accurate, no-reference quality estimation

→ nofu
Gaming QoE and video quality

- several influencing factors: Möller, Schmidt, and Zadtootaghaj [8]
  - video quality factors: content (cgi), encoding (fast),
  - interaction: delay, ...

- objective full-reference metrics: good results: Barman et al. [1, 2, 3]
  - VMAF best; problem: reference usually not available

- for live/adaptive encoding:
  - fast, accurate, no-reference quality estimation

→ nofu
Gaming QoE and video quality

- several influencing factors: Möller, Schmidt, and Zadtootaghaj [8]
  - video quality factors: content (cgi), encoding (fast),
  - interaction: delay, ...

- objective full-reference metrics: good results: Barman et al. [1, 2, 3]
  - VMAF best; problem: reference usually not available

- for live/adaptive encoding:
  - fast, accurate, no-reference quality estimation

→ nofu
Gaming QoE and video quality

- several influencing factors: Möller, Schmidt, and Zadtoothaghaj [8]
  - video quality factors: content (cgi), encoding (fast),
  - interaction: delay, ...

- objective full-reference metrics: good results: Barman et al. [1, 2, 3]
  - VMAF best; problem: reference usually not available

- for live/adaptive encoding:
  - fast, accurate, no-reference quality estimation

→ nofu
Gaming QoE and video quality

- several influencing factors: Möller, Schmidt, and Zadtootaghaj [8]
  - video quality factors: content (cgi), encoding (fast),
  - interaction: delay, ...

- objective full-reference metrics: good results: Barman et al. [1, 2, 3]
  - VMAF best; problem: reference usually not available

- for live/adaptive encoding:
  - fast, accurate, no-reference quality estimation

→ nofu
features:
- $s_i + t_i^M$ [6], $ff_i^l$ [7], staticness$^l$, blockiness$^l$[5],
- cubrow-$\{\text{first, last}\}^M$, cubcol-$\{\text{first, last}\}^M$, blockmotion$^M$[5]

- speedup: 360p center crop of input video
- temporal pooling: 12 feature values per frame
  - first, mean, std, groups $g = [1, 2, 3]$: $\text{mean}_g$, $\text{std}_g$
  - duration independent 108 values per sequence

- ML algorithm: feature selection + RF
- additional no-ref model: brisque+niqe features, similar pipeline
  → Evaluation and used Dataset
nofu – Features and Approach

▶ features:
  - \( si^I + ti^M \) [6], \( fft^I \) [7], staticness' \( s_i \), blockiness'\( b_i \)[5],
  - cubrow-\{first, last\}\( M \), cubcol-\{first, last\}\( M \), blockmotion\( M \)[5]

▶ speedup: 360p center crop of input video
▶ temporal pooling: 12 feature values per frame
  - first, mean, std, groups \( g \) = \([1, 2, 3]\): mean\(_g\), std\(_g\)
  - \( \rightarrow \) duration independent 108 values per sequence

▶ ML algorithm: feature selection + RF
▶ additional no-ref model: brisque+niqe features, similar pipeline
  \( \rightarrow \) Evaluation and used Dataset
nofu – Features and Approach

Video

Feature Extraction
movement, staticness, blockiness, si, ti, ..

Temporal Pooling
mean, std, first mean_g_1, ...3 std_g1, .. 3

Machine Learning Model
Feature Selection + Random Forest

features:
- \( si^I + ti^M \) \([6]\), \( fft^I \) \([7]\), staticness\( ^I \), blockiness\( ^I \)[5],
- \( \text{cubrow-\{first, last\}}^M \), \( \text{cubcol-\{first, last\}}^M \), blockmotion\( ^M \)[5]

- speedup: 360p center crop of input video
- temporal pooling: 12 feature values per frame
  - first, mean, std, groups \( g = [1, 2, 3] \): mean\( _g \), std\( _g \)
  - duration independent 108 values per sequence

- ML algorithm: feature selection + RF

- additional no-ref model: brisque+niqe features, similar pipeline
  → Evaluation and used Dataset
nofu – Features and Approach

features:
- $s_i^l + t_i^M$ [6], $\text{fft}^l$ [7], staticness$^l$, blockiness$^l$[5],
- cubrow-\{first, last\}$^M$, cubcol-\{first, last\}$^M$, blockmotion$^M$[5]

speedup: 360p center crop of input video

temporal pooling: 12 feature values per frame
- first, mean, std, groups $g = [1, 2, 3]$: mean$_g$, std$_g$
- $\rightarrow$ duration independent 108 values per sequence

ML algorithm: feature selection + RF

additional no-ref model: brisque+niqe features, similar pipeline
$\rightarrow$ Evaluation and used Dataset
**nofu – Features and Approach**

- **Feature Extraction per frame**
  - Video
  - Feature Extraction:
    - movement, staticness, blockiness, si, ti, ..
  - Temporal Pooling:
    - mean, std, first
    - mean_g_1, ...3
    - std_g1, .. 3
  - Machine Learning Model:
    - Feature Selection + Random Forest

- **Features**:
  - \(si^I + ti^M\) \([6]\), \(fft^I\) \([7]\), staticness\(^I\), blockiness\(^I\)[5],
  - cubrow-{first,last}\(^M\), cubcol-{first,last}\(^M\), blockmotion\(^M\)[5]

- **Speedup**: 360p center crop of input video

- **Temporal Pooling**: 12 feature values per frame
  - first, mean, std, groups \(g = 1, 2, 3\): \(mean_g\), \(std_g\)
  - \(\rightarrow\) duration independent 108 values per sequence

- **ML algorithm**: feature selection + RF

- **Additional no-ref model**: brisque+niqe features, similar pipeline
  - \(\rightarrow\) Evaluation and used Dataset
nofu – Features and Approach

- **features:**
  - $si^l + ti^M$ [6], $fft^l$ [7], staticness$^l$, blockiness$^l$[5],
  - cubrow-{$first, last}^M$, cubcol-{$first, last}^M$, blockmotion$^M$[5]

- **speedup:** 360p center crop of input video

- **temporal pooling:** 12 feature values per frame
  - $first$, mean, $std$, groups $g = [1, 2, 3]$: $mean_g$, $std_g$
  - → duration independent 108 values per sequence

- **ML algorithm:** feature selection + RF

- **additional no-ref model:** brisque+niqe features, similar pipeline
  → Evaluation and used Dataset
nofu – Features and Approach

Video

Feature Extraction

movement, staticness, blockiness, si, ti, ..

Temporal Pooling

mean, std, first
mean_g_1, ...3
std_g1, .. 3

Machine Learning Model

Feature Selection + Random Forest

► features:
  ○ cubrow-{first,last}$^M$, cubcol-{first,last}$^M$, blockmotion$^M[5]$

► speedup: 360p center crop of input video

► temporal pooling: 12 feature values per frame
  ○ first, mean, std, groups $g = [1, 2, 3]$: $mean_g$, $std_g$
  ○ $\rightarrow$ duration independent 108 values per sequence

► ML algorithm: feature selection + RF

► additional no-ref model: brisque+niqe features, similar pipeline
  $\rightarrow$ Evaluation and used Dataset
features:
- $s_i^l + t_i^M$ [6], $ff_i^l$ [7], staticness$^l$, blockiness$^l$[5],
- cubrow-$\{\text{first, last}\}^M$, cubcol-$\{\text{first, last}\}^M$, blockmotion$^M$[5]

speedup: 360p center crop of input video

temporal pooling: 12 feature values per frame
- $\text{first, mean, std}$, groups $g = [1, 2, 3]$: $\text{mean}_g, \text{std}_g$
- $\rightarrow$ duration independent 108 values per sequence

ML algorithm: feature selection + RF

additional no-ref model:brisque+niqe features, similar pipeline
$\rightarrow$ Evaluation and used Dataset
**nofu – Features and Approach**

- **Video**
- **Feature Extraction**
  - movement, staticness, blockiness, si, ti, ..
- **Temporal Pooling**
  - mean, std, first mean_g_1, ...3
  - std_g1, .. 3
- **Machine Learning Model**
  - Feature Selection + Random Forest

- **Features:**
  - cubrow-{first,last}$^M$, cubcol-{first,last}$^M$, blockmotion$^M [5]$

- **Speedup:** 360p center crop of input video

- **Temporal Pooling:** 12 feature values per frame
  - $first$, $mean$, $std$, groups $g = [1, 2, 3]$: $mean_g$, $std_g$
  - → duration independent 108 values per sequence

- **ML Algorithm:** feature selection + RF

- **Additional no-ref model:** brisque+niqe features, similar pipeline

→ Evaluation and used Dataset
nofu – Features and Approach

- **features:**
  - $s_i + t_i^M$ [6], $\text{fft}_I$ [7], $\text{staticness}_I$, $\text{blockiness}_I$ [5],
  - cubrow-{$\text{first, last}$}$^M$, cubcol-{$\text{first, last}$}$^M$, blockmotion$^M$ [5]

- **speedup:** 360p center crop of input video

- **temporal pooling:** 12 feature values per frame
  - first, mean, std, groups $g = [1, 2, 3]$: mean$_g$, std$_g$
  - $\rightarrow$ duration independent 108 values per sequence

- **ML algorithm:** feature selection + RF

- **additional no-ref model:** brisque+niqe features, similar pipeline
  $\rightarrow$ Evaluation and used Dataset
Evaluation – Dataset

▶ **GamingVideoSET**: Barman et al. [4]:

- 24 full-HD sources, 576 distorted videos, 90 with subjective scores

▶ two main evaluations: 10-fold cross validation and source fold:

- (1) based on VMAF, (2) based on subjective scores

→ MOS prediction
Evaluation – Dataset

GamingVideoSET: Barman et al. [4]:
- 24 full-HD sources, 576 distorted videos, 90 with subjective scores

Two main evaluations: 10-fold cross validation and source fold:
- (1) based on VMAF, (2) based on subjective scores

→ MOS prediction
Evaluation – Dataset

- **GamingVideoSET**: Barman et al. [4]:
  - 24 full-HD sources, 576 distorted videos, 90 with subjective scores

- Two main evaluations: 10-fold cross validation and source fold:
  - (1) based on VMAF, (2) based on subjective scores

→ MOS prediction
Evaluation – Dataset

GamingVideoSET: Barman et al. [4]:

- 24 full-HD sources, 576 distorted videos, 90 with subjective scores

Two main evaluations: 10-fold cross validation and source fold:

- (1) based on VMAF, (2) based on subjective scores

→ MOS prediction
Evaluation – Dataset

GamingVideoSET: Barman et al. [4]:

- 24 full-HD sources, 576 distorted videos, 90 with subjective scores

Two main evaluations: 10-fold cross validation and source fold:

- (1) based on VMAF, (2) based on subjective scores
  → MOS prediction
Evaluation – MOS prediction

Pearson (P), Spearman (S), Kendall (K) and RMSE

- nofu > brisque + niqe > vmaf > ssim

- Source video fold evaluation: nofu > brisque + niqe

→ Conclusion
Evaluation – MOS prediction

Pearson (P), Spearman (S), Kendall (K) and RMSE

- nofu > brisque+niqe > VMAF > SSIM
- Source video fold evaluation: nofu > brisque+niqe

→ Conclusion
Evaluation – MOS prediction

Pearson (P), Spearman (S), Kendall (K) and RMSE

- nofu > brisque + niqe > VMAF > ssim

- Source video fold evaluation: nofu > brisque + niqe

→ Conclusion
Conclusion, Summary and Future Work

- introduced **nofu**: no-reference model for gaming video quality
  - **features**: quality-related and gaming-specific
  - **temporal pooling + 360p center crop**
  - machine learning based
- evaluation using GamingVideoSET [4]
  - **nofu** outperforms other no-ref models + VMAF
  - per source fold: promising results
- open and next steps:
  - include delay/latency, bitstream features, combine **nofu** + **brisque** + **niqe**
  - use features/approach for different tasks
Conclusion, Summary and Future Work

- introduced **nofu**: no-reference model for gaming video quality
  - **features**: quality-related and gaming-specific
  - **temporal pooling + 360p center crop**
  - machine learning based

- evaluation using GamingVideoSET [4]
  - **nofu** outperforms other no-ref models + VMAF
  - per source fold: promising results

- open and next steps:
  - include delay/latency, bitstream features, combine **nofu** + brisque + niqe
  - use features/approach for different tasks
Conclusion, Summary and Future Work

- introduced **nofu**: no-reference model for gaming video quality
  - **features**: quality-related and gaming-specific
  - **temporal pooling + 360p center crop**
    - machine learning based

- evaluation using GamingVideoSET [4]
  - **nofu** outperforms other no-ref models + VMAF
  - per source fold: promising results

- open and next steps:
  - include delay/latency, bitstream features, combine **nofu**+brisque+niqe
  - use features/approach for different tasks
Conclusion, Summary and Future Work

- introduced **nofu**: no-reference model for gaming video quality
  - **features**: quality-related and gaming-specific
  - **temporal pooling + 360p center crop**
  - machine learning based

- evaluation using GamingVideoSET [4]
  - **nofu** outperforms other no-ref models + VMAF
  - per source fold: promising results

- open and next steps:
  - include delay/latency, bitstream features, combine **nofu**+brisque+niqe
  - use features/approach for different tasks
Conclusion, Summary and Future Work

- introduced **nofu**: no-reference model for gaming video quality
  - **features**: quality-related and gaming-specific
  - **temporal pooling + 360p center crop**
  - machine learning based

- **evaluation using GamingVideoSET [4]**
  - **nofu** outperforms other no-ref models + VMAF
  - per source fold: promising results

- **open and next steps:**
  - include delay/latency, bitstream features, combine **nofu** + **brisque** + **niqe**
  - use features/approach for different tasks
Conclusion, Summary and Future Work

- introduced **nofu**: no-reference model for gaming video quality
  - **features**: quality-related and gaming-specific
  - **temporal pooling + 360p center crop**
  - machine learning based

- evaluation using GamingVideoSET [4]
  - **nofu** outperforms other no-ref models + VMAF
  - per source fold: promising results

- open and next steps:
  - include delay/latency, bitstream features, combine nofu+brisque+nique
  - use features/approach for different tasks
Conclusion, Summary and Future Work

- introduced **nofu**: no-reference model for gaming video quality
  - **features**: quality-related and gaming-specific
  - **temporal pooling + 360p center crop**
  - machine learning based

- evaluation using GamingVideoSET [4]
  - **nofu** outperforms other no-ref models + VMAF
  - per source fold: promising results

- open and next steps:
  - include delay/latency, bitstream features, combine nofu + brisque + niqe
  - use features/approach for different tasks
Conclusion, Summary and Future Work

- introduced **nofu**: no-reference model for gaming video quality
  - **features**: quality-related and gaming-specific
  - **temporal pooling + 360p center crop**
  - machine learning based

- evaluation using GamingVideoSET [4]
  - **nofu** outperforms other no-ref models + VMAF
  - per source fold: promising results

- open and next steps:
  - include delay/latency, bitstream features, combine **nofu**+brisque+niqe
  - use features/approach for different tasks
Conclusion, Summary and Future Work

- introduced **nofu**: no-reference model for gaming video quality
  - **features**: quality-related and gaming-specific
  - **temporal pooling + 360p center crop**
  - machine learning based

- evaluation using GamingVideoSET [4]
  - **nofu** outperforms other no-ref models + VMAF
  - per source fold: promising results

- open and next steps:
  - include delay/latency, bitstream features, combine **nofu** + brisque + niqe
  - use features/approach for different tasks
Conclusion, Summary and Future Work

- introduced **nofu**: no-reference model for gaming video quality
  - **features**: quality-related and gaming-specific
  - **temporal pooling + 360p center crop**
  - machine learning based

- evaluation using GamingVideoSET [4]
  - **nofu** outperforms other no-ref models + VMAF
  - per source fold: promising results

- open and next steps:
  - include delay/latency, bitstream features, combine **nofu**+brisque+niqe
  - use features/approach for different tasks
Thank you for your attention

...... are there any questions?
References


References II


