

IGVQM Current Status & Results

JEG-Hybrid (Joint Effort Group)

Enrico Masala

Politecnico di Torino, Turin, Italy

VQEG May 2025 Meeting

Implementer's Guide to Video Quality Metrics

- More info on JEG-Hybrid website: <https://vqegjeg.github.io/jeg-hybrid/igvqm>
- Main idea: use a large set of subjectively-annotated datasets to offer standard guidelines in correctly using VQMs
 - Based on open data
 - Reproducible experiments

Database name	Contributor	Year	# of sources	# of PVS	# of opinion scores	Methodology	Scale (discrete/continuous)	Scoring
FOO	BAR	2008	30	300	9000	ACR	Discrete	
VQEG HD 1 to 5	VQEG	2010				ACR-HR	Discrete	
ITS4S	NTIA/ITS	2018				ACR	Discrete	
ITS4S2	NTIA/ITS	2019				ACR	Discrete	
ITS4S3	NTIA/ITS	2019				ACR	Discrete	
ITS4S4	NTIA/ITS	2020				ACR	Discrete	
CCRIQ	Intel, ITS, U. Ghe	2016				ACR	Discrete	
Public Safety #1	NTIA/ITS	2005				ACR-HR	Discrete	
Public Safety #2	NTIA/ITS	2006				ACR-HR	Discrete	
T1A1	ATIS	1996				DSIS	Discrete	
GamingVideoSET	KU	2018	6	90	2250	ACR	Discrete	
KUGVD	KU	2018	6	90	1530	ACR	Discrete	
AVT-VQDB-UHD-1	TU Ilmenau	2019	12	756	around 18,000 (t	ACR	Discrete	
AGH/NTIA/Dolby	AGH, NTIA, Dolt	2018				ACR-HR	Discrete	

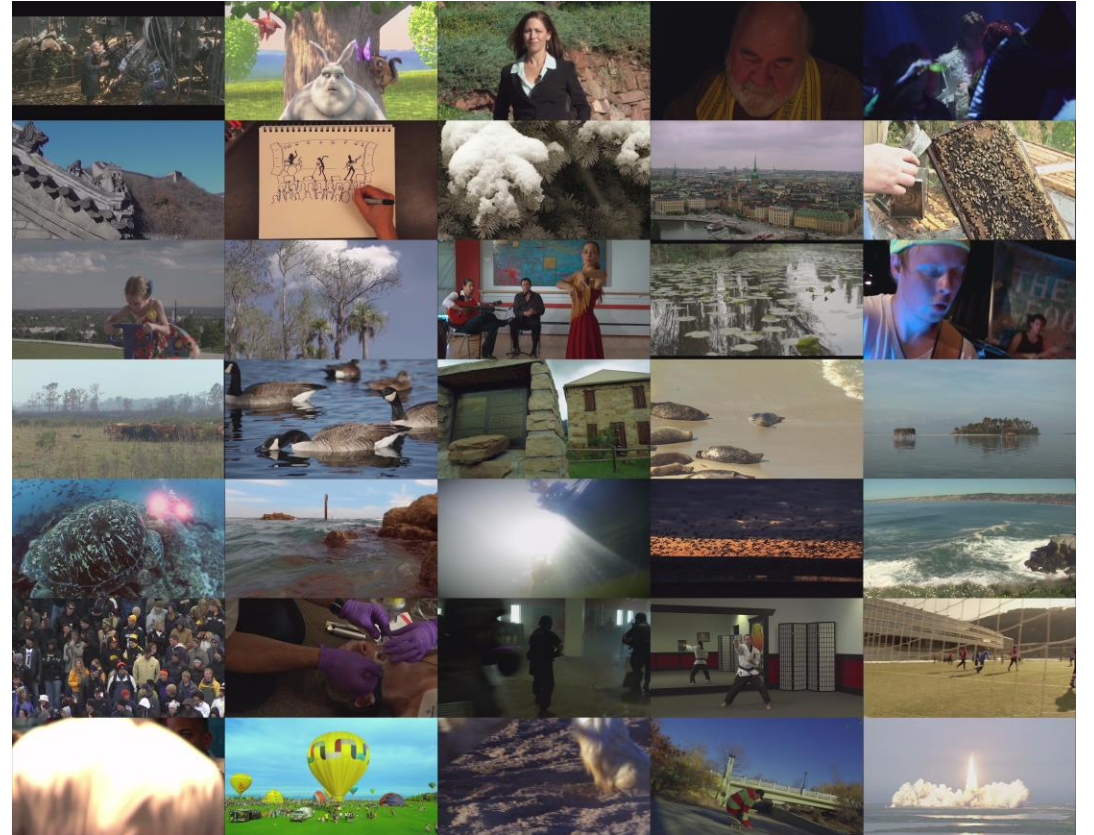
https://docs.google.com/spreadsheets/d/1wYVjaqKrSB0z7D2eJlAd1g6y_831IF7TH6GhAKdKxA/edit

Selected Datasets

Dataset	PVS Resolution (max → min)	# PVS	# SRC	Codecs	Method	Notes
ITS4S	720p	514	514	?	ACR	
KUGVD	1080p → 480p	90	6	AVC	ACR	Gaming content
GamingVideoSET	1080p → 480p	90	6	AVC	ACR	Gaming content
AGH-NTIA-Dolby	720p	224	84	AVC, MPEG2	ACR-HR	Discarded 6 SRC-PVS pairs due to temporal misalignment
AVT-VQDB-UHD-1	2160p → 360p 60 → 30 → 15 <i>fps</i>	756	20	AVC, HEVC, VP9	ACR	Some seq are the same but shorter version
TOTAL		1674	630			

Introduction to Results

- **NB: Still work-in-progress**
- A few results are presented here to give ideas about the direction and receive suggestions / feedbacks
- Most of the results in next slides are from the ITS4S dataset

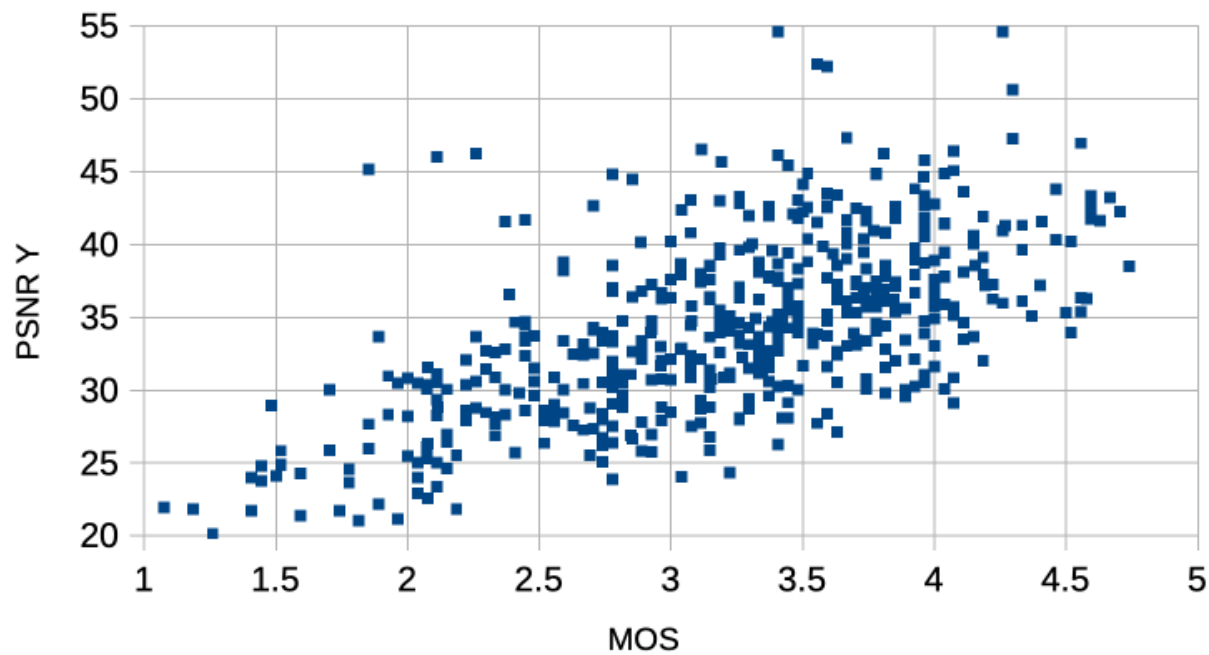


Random screenshots from the ITS4S dataset (514 PVSs)

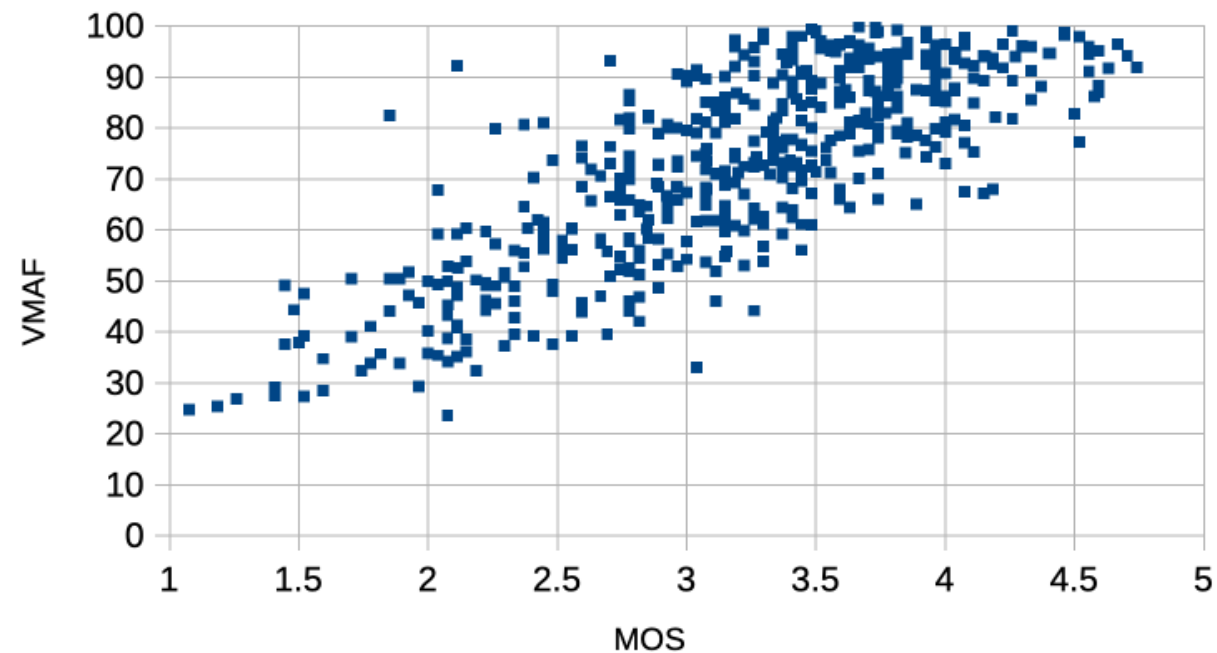
Metrics vs MOS

- ITS4S dataset

PSNR Y PLCC: 0.6161



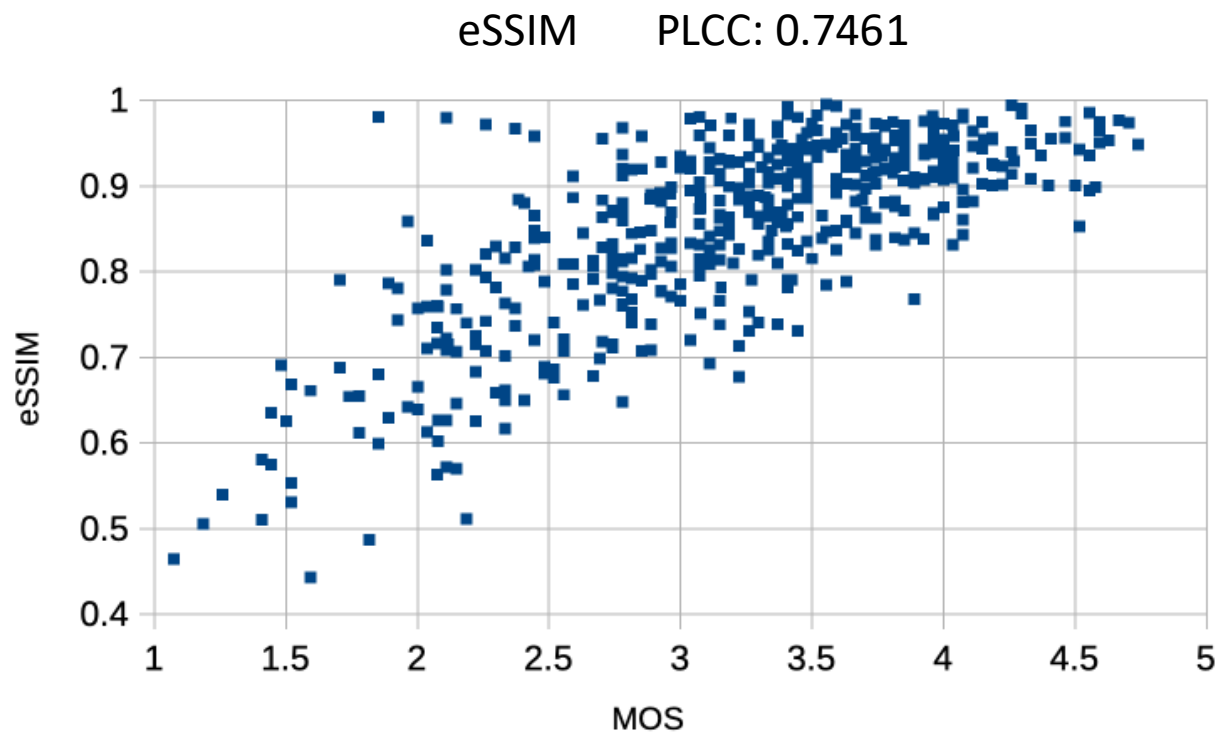
VMAF PLCC: 0.7960



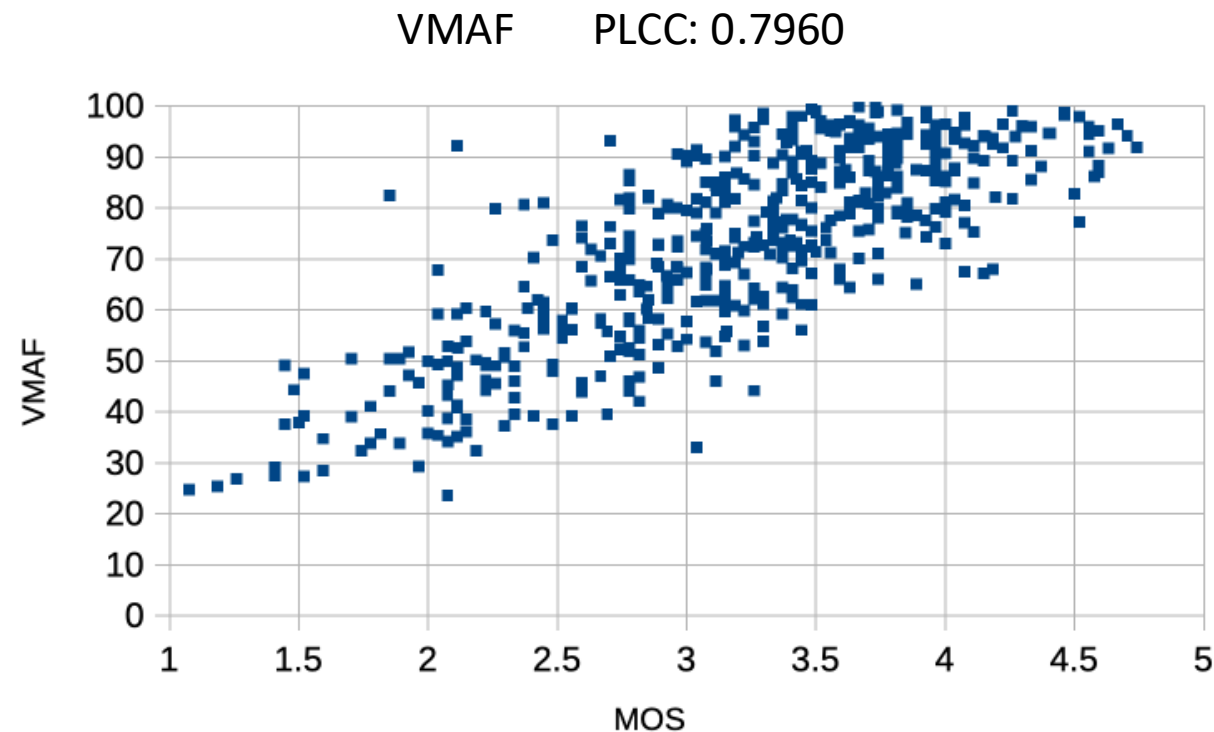
VMAF version v.0.6.1

Metrics vs MOS

- ITS4S dataset



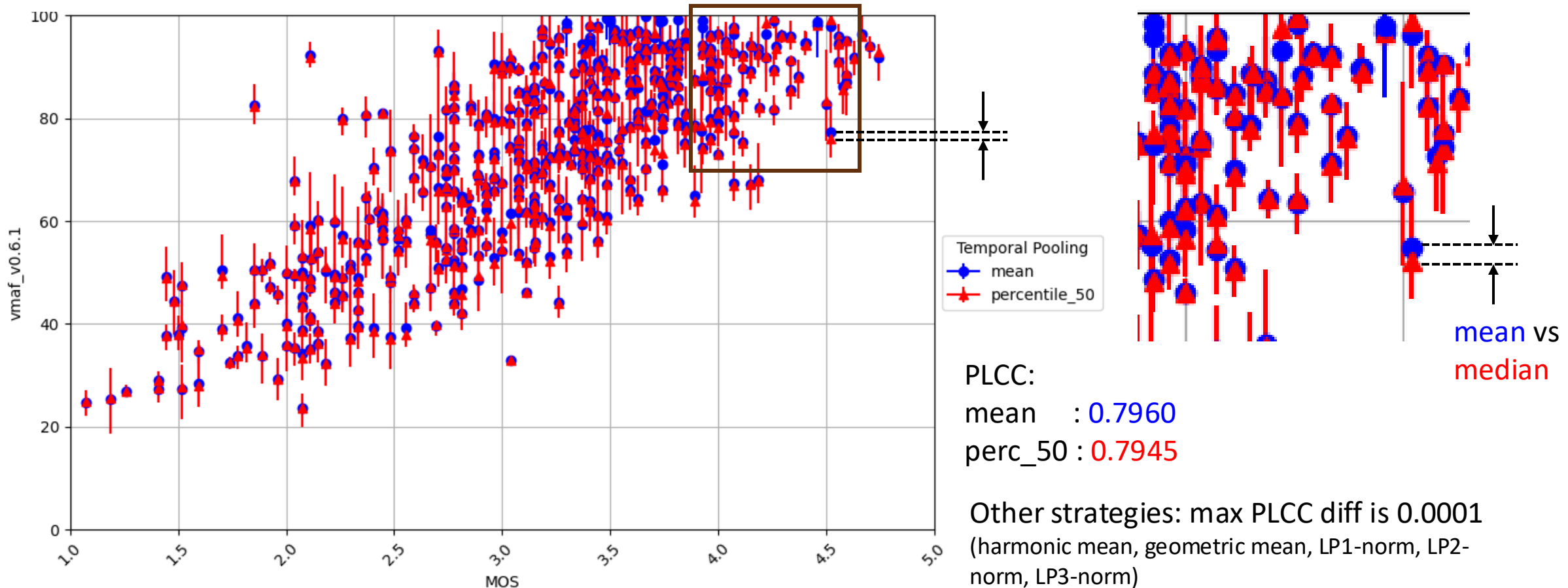
(eSSIM default parameters: window size 8, window stride 4, Minkowski pooling 3, mode 2: Performance)



VMAF version v.0.6.1

Temporal Aggregation Strategies

- ITS4S dataset: **mean** of VMAF vs **median (50-th percentile)** (also showing **bar** for 5-95th perc.)



PLCC:

mean : 0.7960

perc_50 : 0.7945

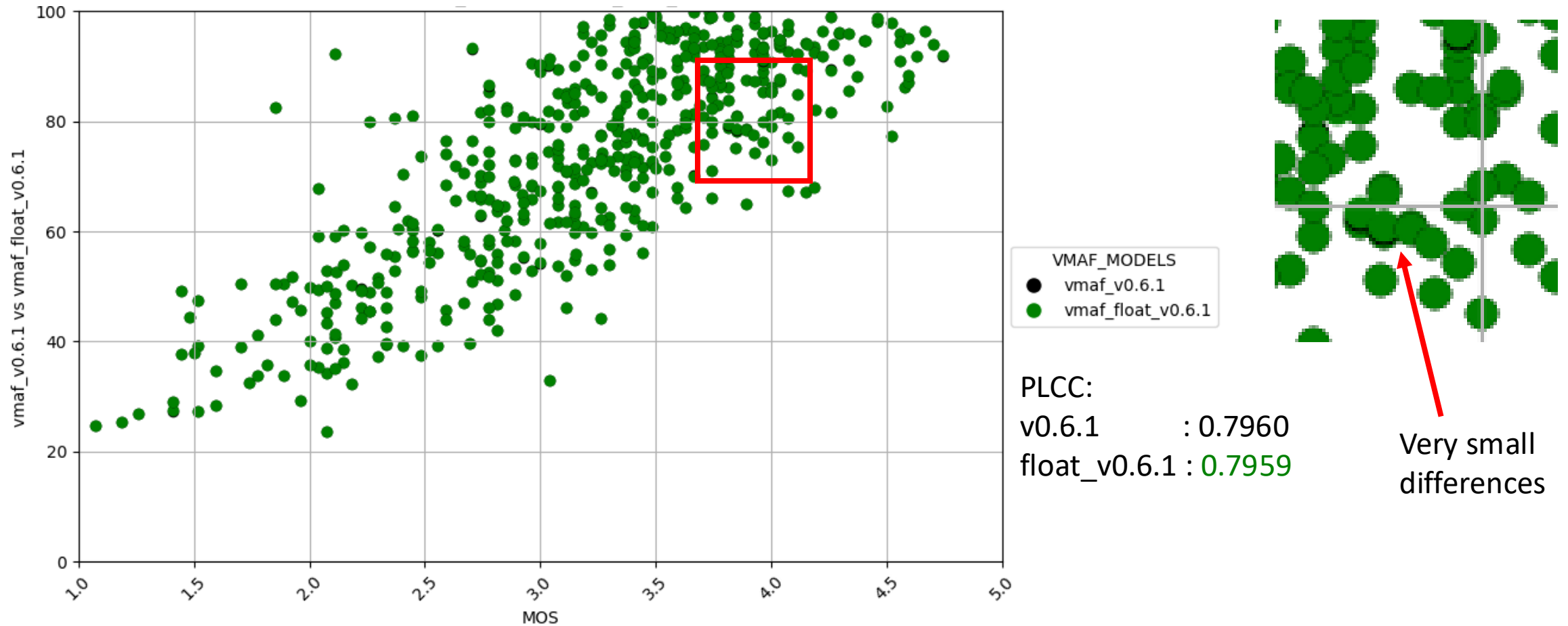
Other strategies: max PLCC diff is 0.0001

(harmonic mean, geometric mean, LP1-norm, LP2-norm, LP3-norm)

<https://github.com/Netflix/vmaf/blob/master/python/vmaf/tools/stats.py>

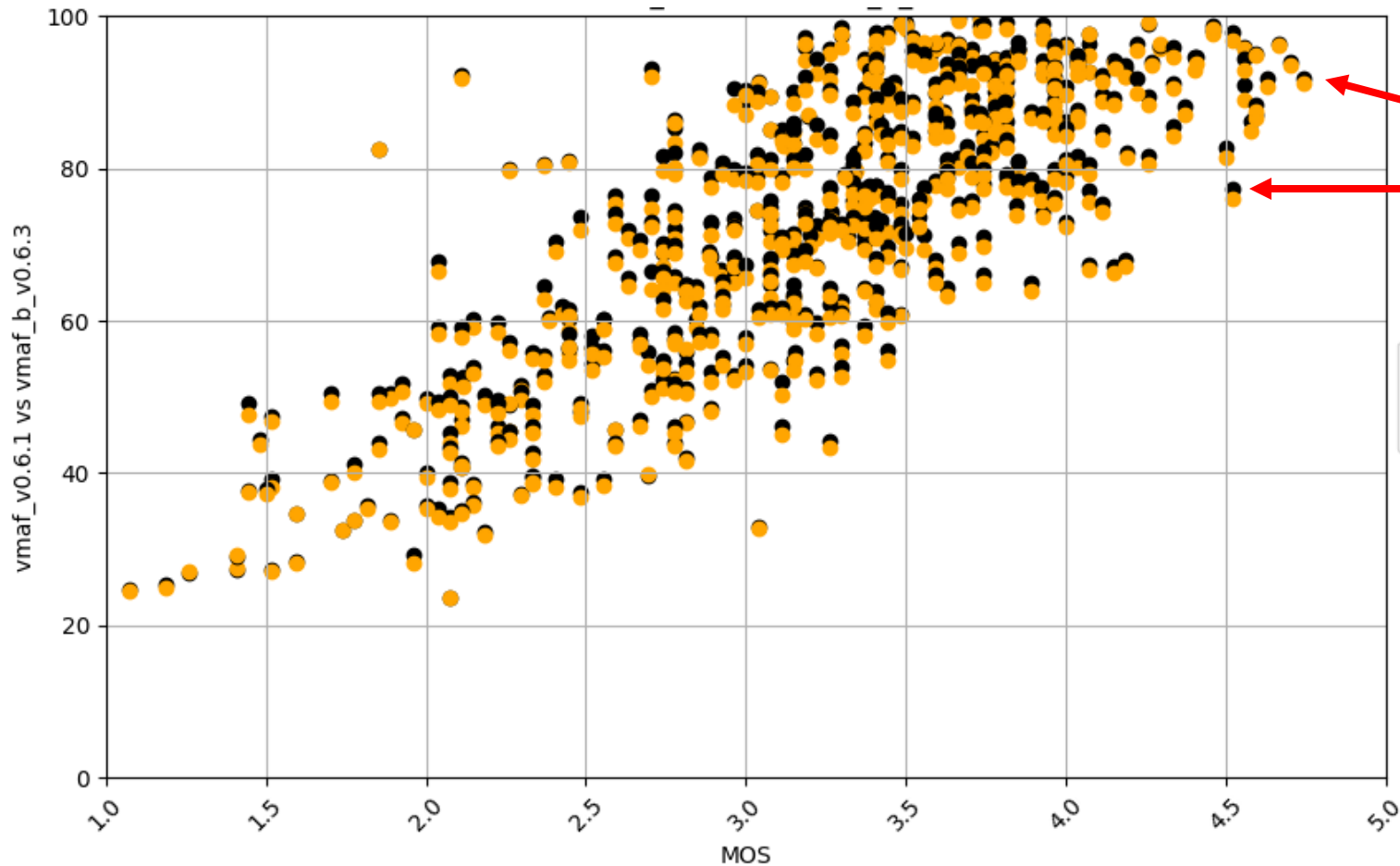
Results of Different VMAF Implementations

- ITS4S dataset: VMAF vs VMAF floating point implementation



Different VMAF Models

- ITS4S dataset: VMAF version 0.6.1 vs 0.6.3



Sometimes results
are different,
sometimes less

PLCC:

v0.6.1 : 0.7960

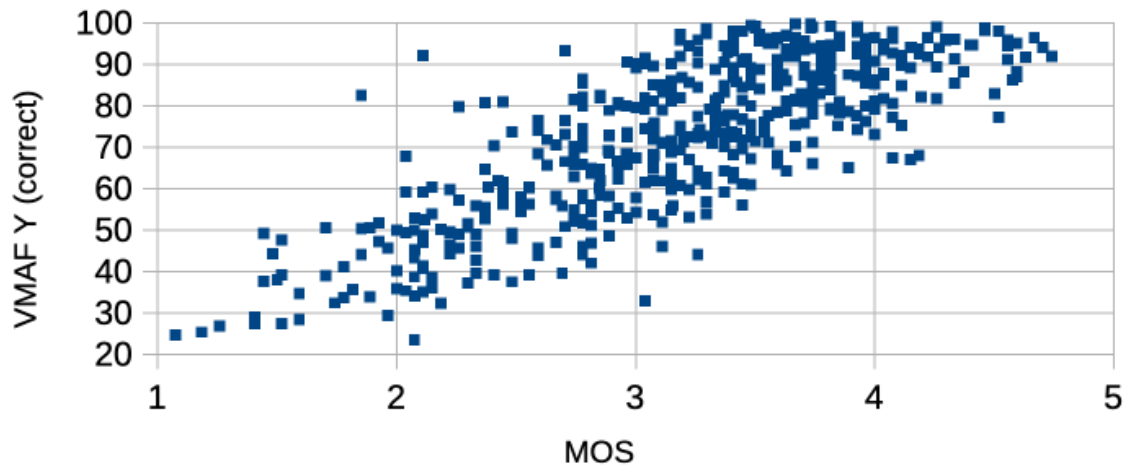
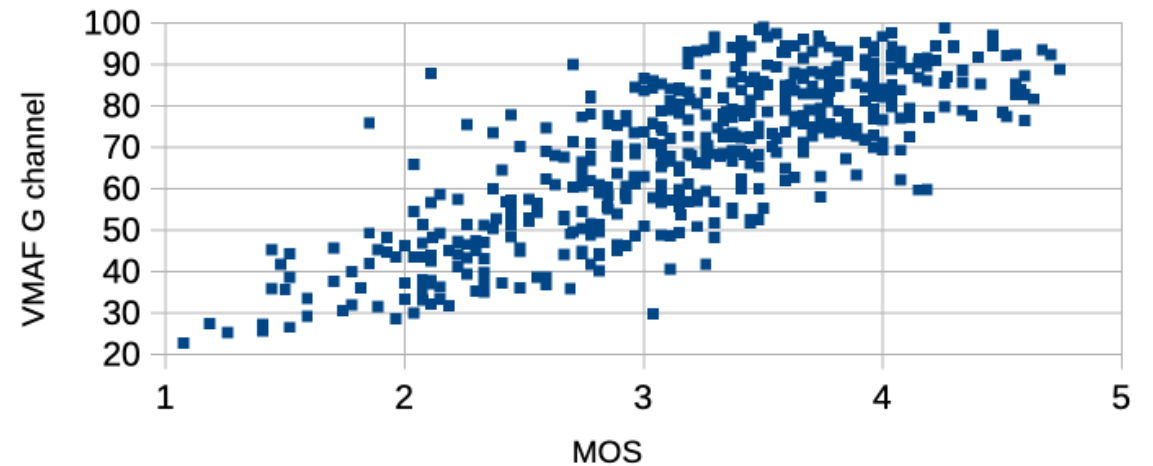
b_v0.6.3 : 0.7953

Effect of Common Mistakes

- What happens if:
 - Computing VMAF in wrong colorspace
 - Using inappropriate model for content resolution
 - Using different SSIM implementations

VMAF in RGB colorspace

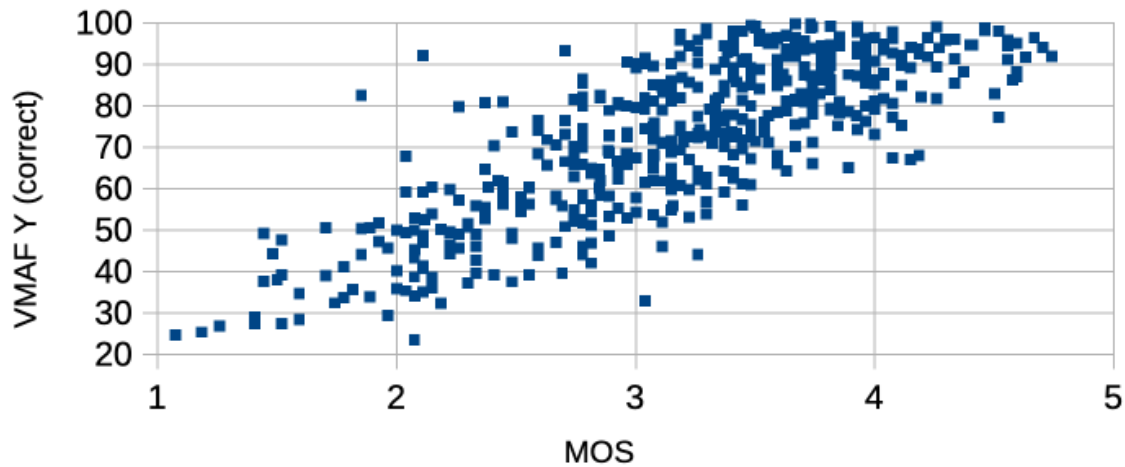
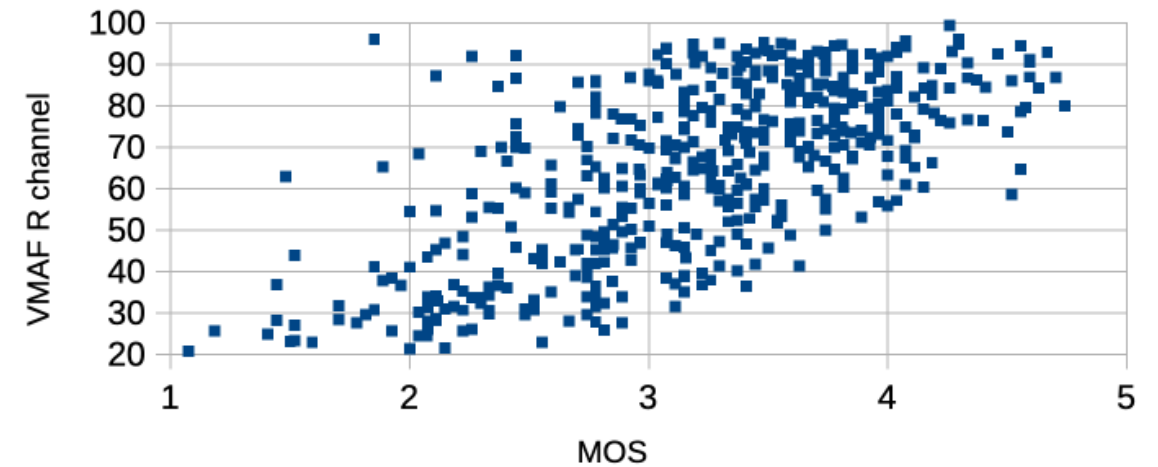
ITS4S dataset, MOS to VMAF PLCC:
0.7960 (Y) 0.7790 (G) -0.0170 PLCC



- If using G instead of Y, PLCC change is limited: about -0.02

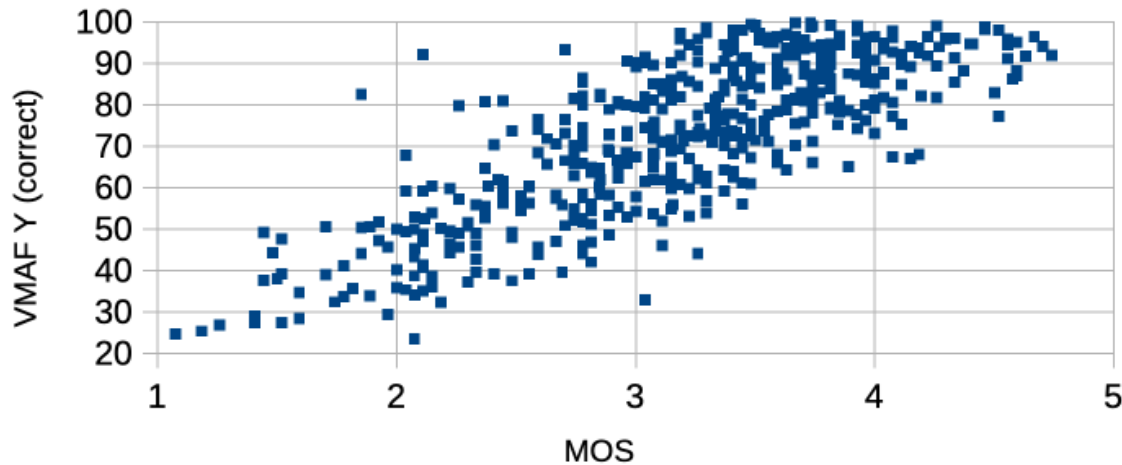
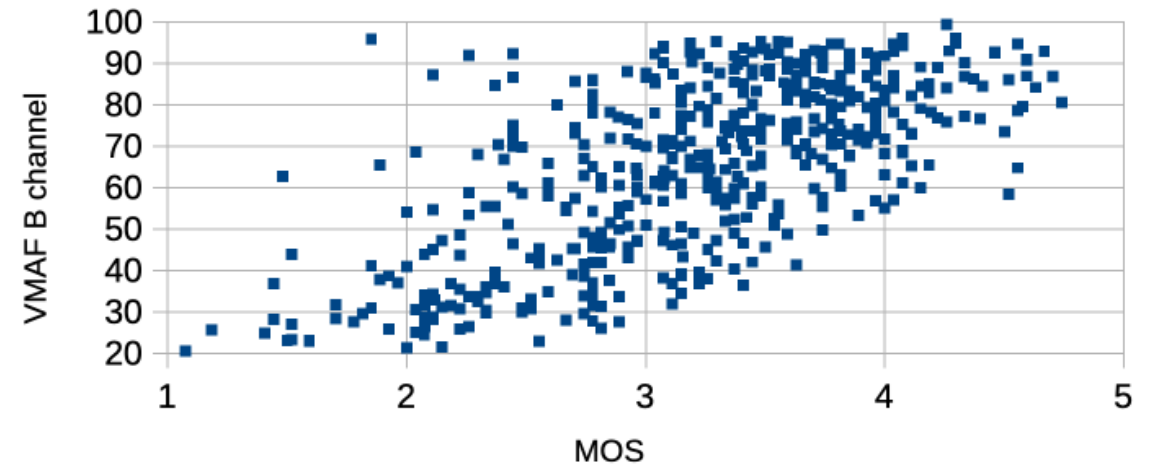
VMAF in RGB colorspace

ITS4S dataset, MOS to VMAF PLCC:
0.7960 (Y) 0.6937 (R) -0.1023 PLCC



VMAF in RGB colorspace

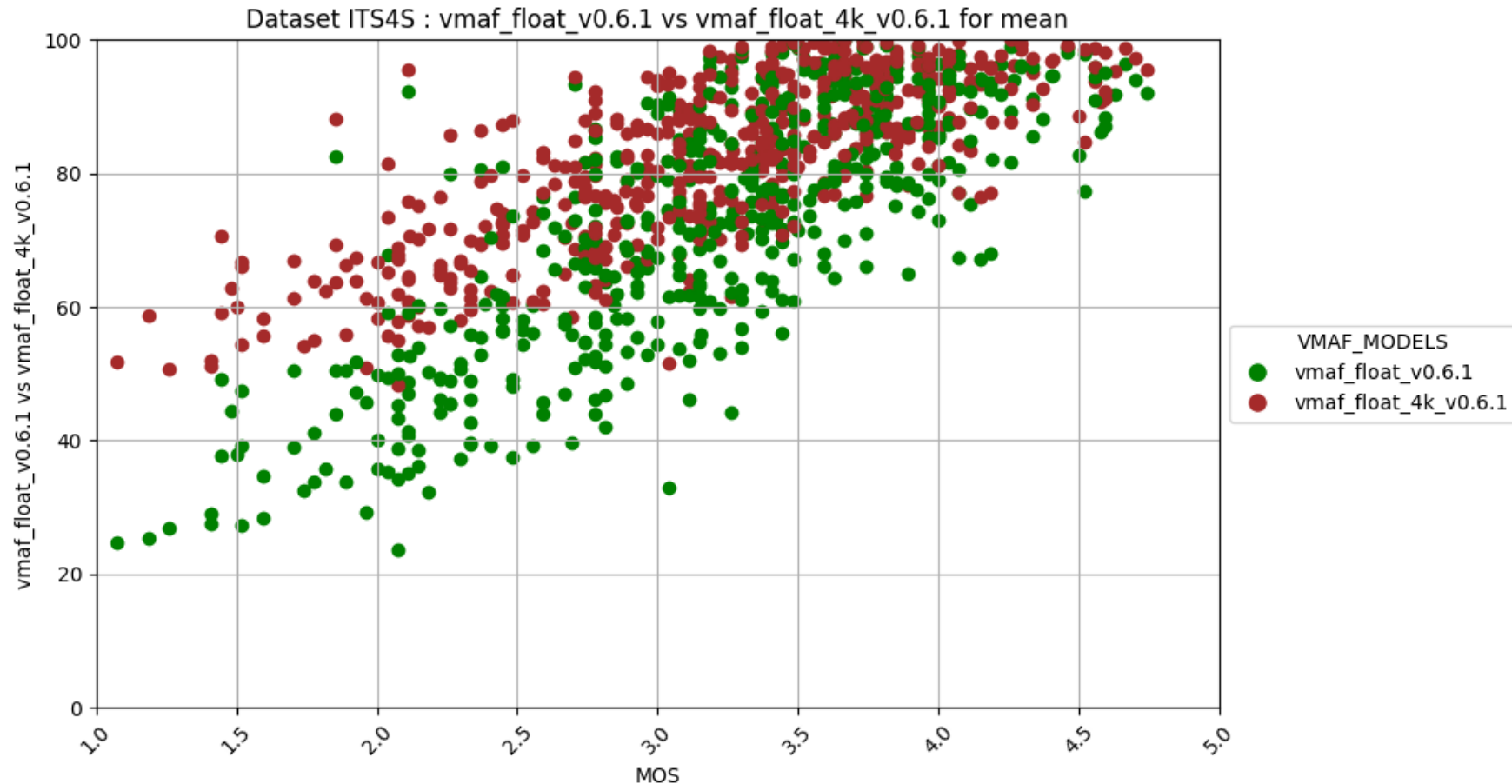
ITS4S dataset, MOS to VMAF PLCC:
0.7960 (Y) 0.6941 (B) -0.1019 PLCC



- If using R and B instead of Y, PLCC may change significantly: -0.10

Wrong VMAF Model (for resolution)

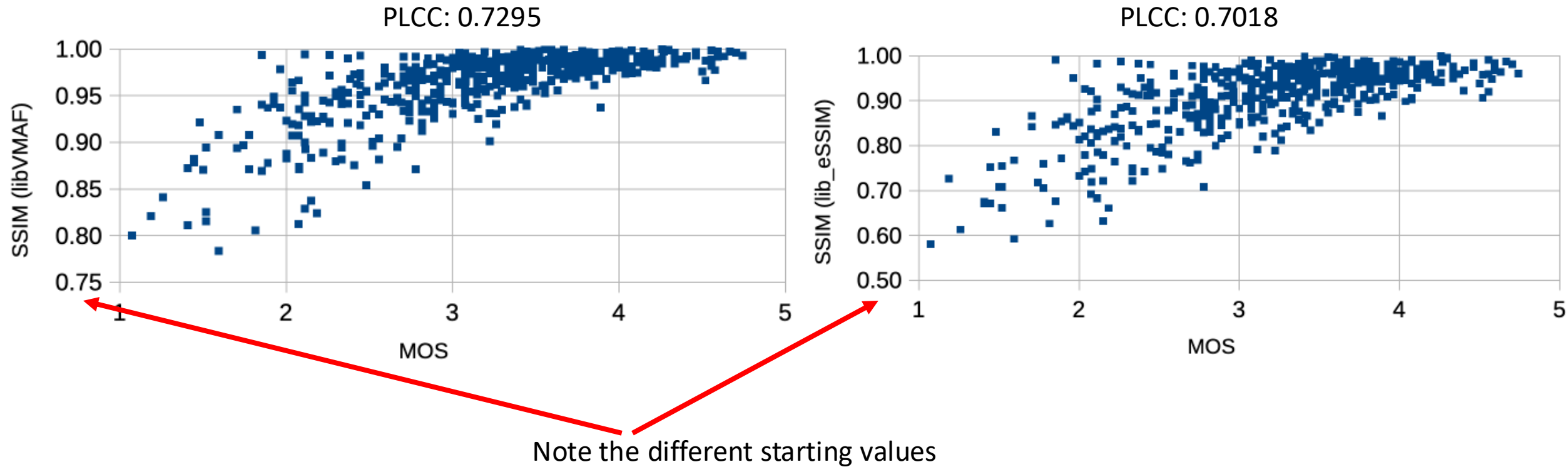
- ITS4S dataset: VMAF 4k (red) vs VMAF “basic” can overestimate a lot



PLCC:
float : 0.7960
float_4k: 0.7811

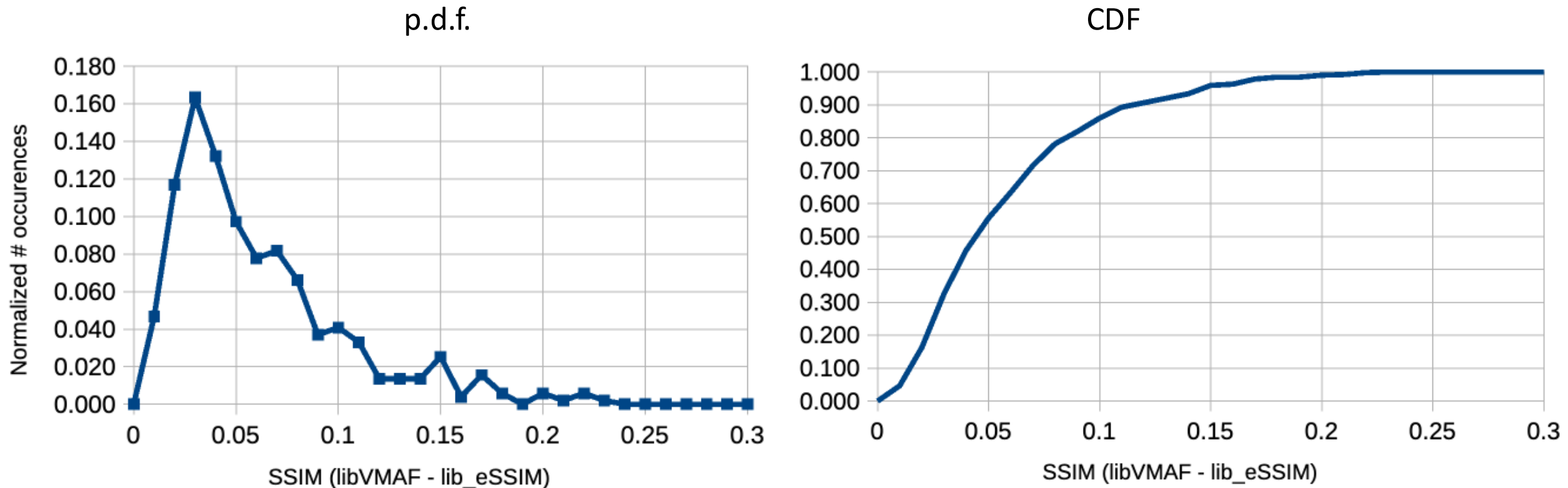
SSIM: Effect of Different Implementations

- ITS4S dataset, **SSIM** from: libVMAF (Netflix) & from essim software (Facebook Research)



SSIM: Effect of Different Implementations

- ITS4S dataset, **SSIM** from: libVMAF (Netflix) & eSSIM (Facebook Research)



Current Software Status

- Implemented code (based on docker/podman for maximum reproducibility) is publicly available on GitHub:

https://github.com/emasala/IGVQM_analysis

- Still requiring many refinements / refactoring
- However, all **MOS** values from the 5 considered datasets have been converted and made available in the same JSON format:

https://github.com/emasala/IGVQM_analysis/tree/main/Mos

May allow easier usage/processing of MOS values also outside this project

Next Steps

- Finish analysis for all datasets, report significant results in next meetings
- Include computational complexity (at least in the form of running time)
- Create colab-style pages for autonomous experimentation with graphs (?)
- Write a first draft of IGVQM whitepaper / guidelines
- Academic publication (?)
 - Some results about temporal pooling, impact of upscaling algorithms on VMAF (not tested yet), etc.

Comments / Proposals / Ideas

- **Welcome now!**



- Join our biweekly conference call (next: **May 12, 2025, 4pm CET/CEST**)
- References:
 - **JEG-Hybrid website** <https://vqegjeg.github.io/jeg-hybrid/>
 - Including an up-to-date publication section with immediate download link for each one
 - **IGVQM** subsection: <https://vqegjeg.github.io/jeg-hybrid/igvqm>
 - VQEG-JEG mailing list: jeg@vqeg.org (<https://vqeg.org/email-reflectors.aspx>)