Immersive video challenges and some AI solutions

Stéphane Coulombe

2023-05-24

ÉTS
Le génie pour l'industrie

ILLS
International Laboratory on Learning Systems

Immersive video challenges and some AI solutions

1. What is immersive video?
2. What are the challenges?
3. Some AI solutions.

École de technologie supérieure

- Some facts and statistics:
  - Located downtown Montreal, ÉTS has over **11,000 students.**
  - Nearly **2,800 at graduate level** (684 Ph.D. level).
  - 60% of **research** activities **in collaboration with industry.**
  - 28 groups of researchers, 1000 publications/year.
  - Since 2015, **ÉTS is 2nd in Canada** in the number of awarded engineering diplomas, according to Ingénieurs Canada.
  - More than **3,500 internships** carried out annually in 1,200 companies.
  - A thousand employees, including more than **260 professors and lecturers.**

Stéphane Coulombe

- Research Interests
  - Video processing and communications
- Industrial / academic experience:
  - 1996-1999: Bell Northern Research (Nortel)
    - Wireless Speech Group (Member of Scientific staff)
  - 1999-2004: Nokia Research Center (Dallas)
    - Visual Communications Laboratory (Senior Research Engineer / Research Program Manager)
  - Since Oct. 2004: ÉTS, Software and IT Dept. (Professor)
- Several industrial collaborations:
  - From 2009 to 2018, Chairholder of the Vantrix Industrial Research Chair in Video Optimization
  - Since 2021, Co-Chairholder of the Summit Tech Industrial Research Chair in Interactive Immersive Video

ÉTS Le génie pour l'industrie

ILLS International Laboratory on Learning Systems

1. What is immersive video?

What is immersive video?

- Several types of immersive experiences:
  - They include *Augmented*, *Virtual* and *Mixed* Reality (AR, VR, MR); each with unique purposes and capabilities.
  - Can involve the user's multiple sensory modalities, including visual, sound, motion, pressure, and smell.
  - Here, we focus on 360-degree video with 3DoF (head) and 6DoF (head+body).

What is immersive video?

- Applications:
  - Virtual tourism (e.g., museum and historical site visits)
  - Participation in a live concert from home
  - Real estate (virtual home visits)
  - Virtual shopping

360-degree video processing chain

Capture

Stitch

Projection

Compression

Transmission

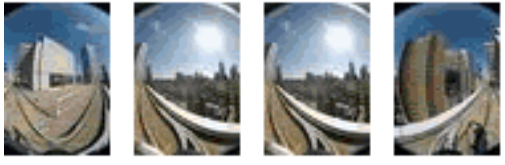
2D/3D

Live/offline

Equirectangular, Sphere…

HEVC/VVC Tiling

Broadcast
Unicast



H.265 MPEG-4/AVC

H.266 VVC

AI

Display

Rendering

De-compression

Reception

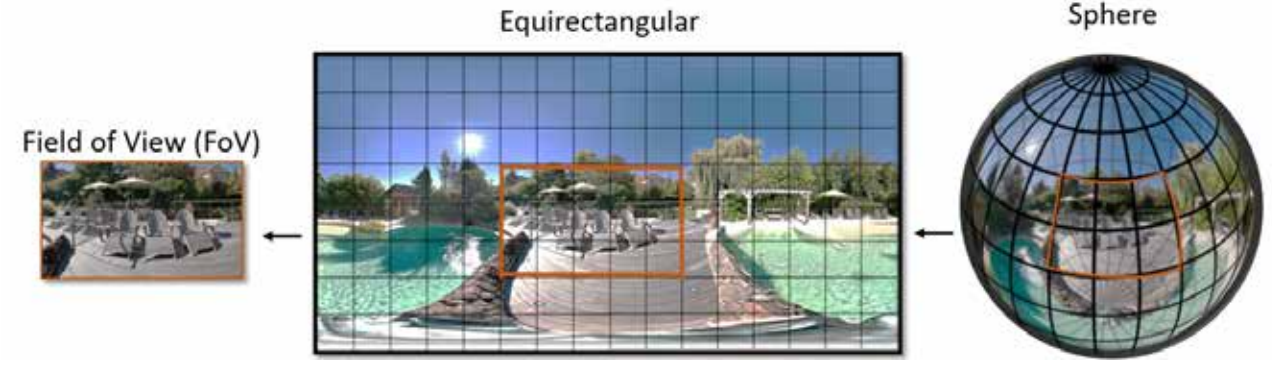
H.265 MPEG-4/AVC

H.266 VVC

Head orientation

360-degree video projection

- The whole visual field can be modeled by a sphere.
- The viewer only sees a part of that sphere at any given time.
- The visual content covered by the sphere needs to be transmitted as a video for compression… Therefore, a projection needs to occur…



E.g. Spherical to equirectangular (ERP)

Anahita Mahzari, Afshin Taghavi Nasrabadi, Aliehsan Samiei, and Ravi Prakash. 2018. FoV-Aware Edge Caching for Adaptive 360° Video Streaming. In Proceedings of the 26th ACM international conference on Multimedia (MM '18). Association for Computing Machinery, New York, NY, USA, 173–181.

360-degree video projection

- The projected image contains the whole 360-degree view

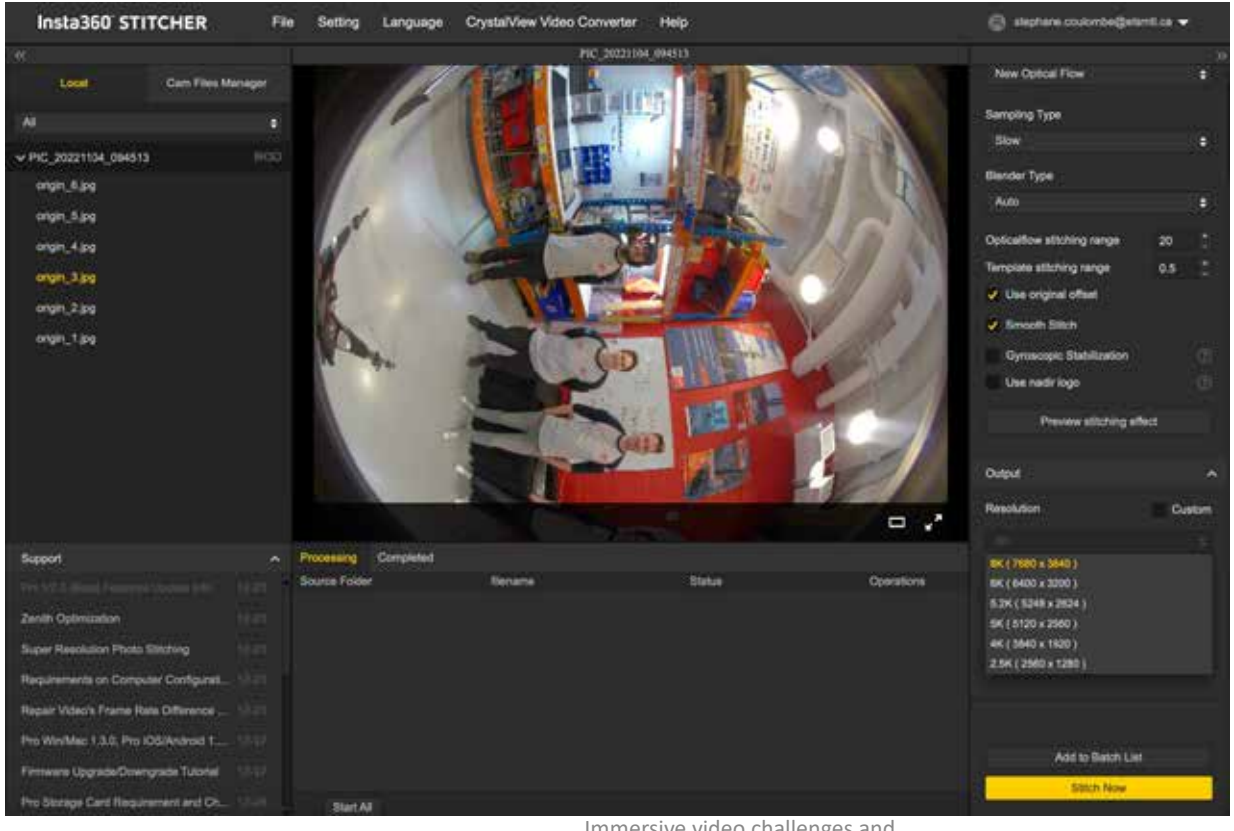
E.g.: YouTube VR channel (360-degree video):

Transmitted image

Displayed images (vue control)

https://vr.youtube.com/

Insta360 Stitcher



Insta360 Stitcher

2. What are the challenges?

Immersive video challenges

1. Immersive video (IV) requires a lot of bit rate
   - E.g. YouTube Channel sends **compressed** 4K ERP videos.
   - But the human field of view (FOV) represents about 1/5 of the sphere.
     - Waste of the transmitted information.
     - The resolution within the FOV is very small (less than HD) → low visual quality
   - Good immersive experiences require 8K to 12K resolution at 60fps!
     - 4K HEVC requires 45-60 Mbps in SDR and 60-75 Mbps for HDR.
     - Bit rate increases with 3D-360 video (stereoscopic vision).

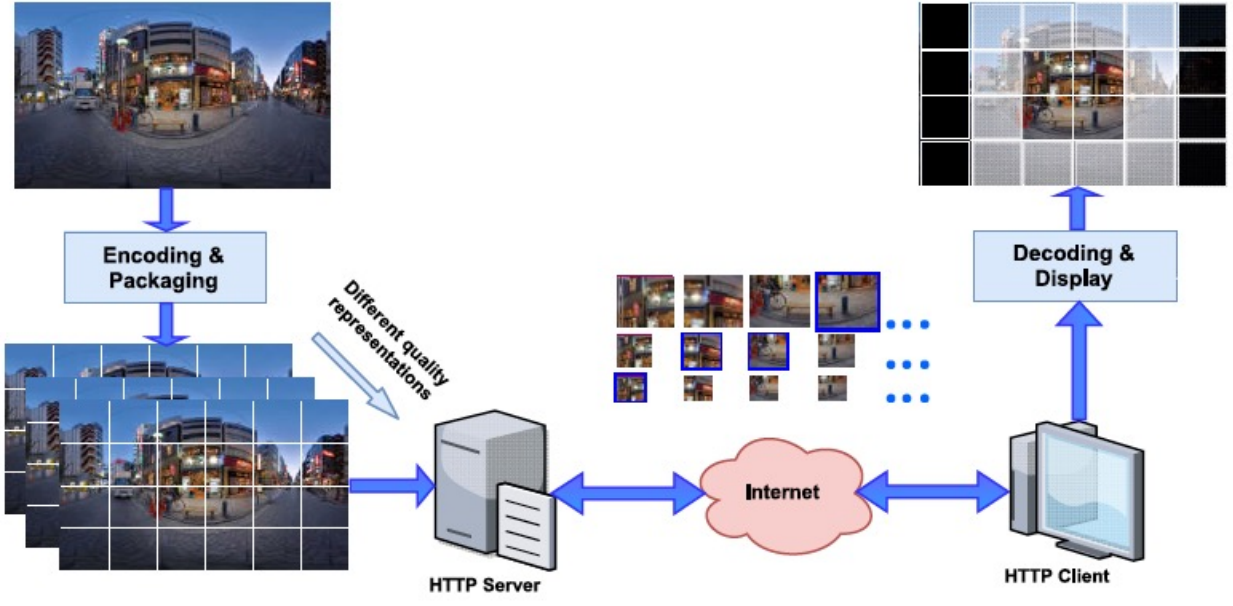
Solutions:

- Highly efficient compression (AI ?)→ computational complexity / latency.
- Send only the visual info inside the FOV → need to know head position.

Immersive video challenges

Solutions:

- Use of tiling:



D. H. Nguyen, M. Nguyen, N. P. Ngoc and T. C. Thang, "An adaptive method for low-delay 360 VR video streaming over HTTP/2," *2018 IEEE Seventh International Conference on Communications and Electronics (ICCE)*, Hue, Vietnam, 2018, pp. 261-266, doi: 10.1109/CCE.2018.8465722.

Immersive video challenges

2. IV requires low latency
   - We want about 20 ms or less between head motion and displayed image.
     - Otherwise, the user gets cybersickness.
   - If we use tiling, we need to know at each moment the head position
     - Can we get this information and send the customized content in time?

Solution:

- Predict accurately the head position/orientation (AI can help!).

Immersive video challenges

3. Visual quality
   - Compression can impact visual quality
     - Compromise between *bitrate – visual quality – computational complexity*.
   - How do we evaluate the quality?
     - Many “reliable” **objective** visual quality metrics for conventional video.
     - A high-interest topic of research in IV.
   - Transmission errors or delays
     - We may not have time for retransmissions. Try to recover damaged content.

Solutions:

- New objective visual quality metrics (AI can help!)
- Error concealment (AI can help!)

3. Some AI solutions.

LSTM-based Viewpoint Prediction for Multi-Quality Tiled Video Coding in Virtual Reality Streaming [R1]

- We developed a sequence-to-sequence predictive model using an LSTM encoder-decoder.
- The predictive model takes the user's viewpoint position history as a sequence and predicts the future viewpoint positions as a sequence as well.
- The encoder takes a sliding window of *M* frames' features (yaw or pitch angles) and a time distributed dense layer, implemented after the decoder, outputs a prediction window of *N* frames' features (yaw or pitch angles).
- We thus obtain the predicted viewpoint for future frames ranging from the next one up to the *N-th* one.

[R1] M. Jamali, S. Coulombe, A. Vakili and C. Vazquez, "LSTM-Based Viewpoint Prediction for Multi-Quality Tiled Video Coding in Virtual Reality Streaming," *2020 IEEE International Symposium on Circuits and Systems (ISCAS)*, Seville, Spain, 2020, pp. 1-5, doi: 10.1109/ISCAS45731.2020.9180528.

LSTM-based Viewpoint Prediction for Multi-Quality Tiled Video Coding in Virtual Reality Streaming

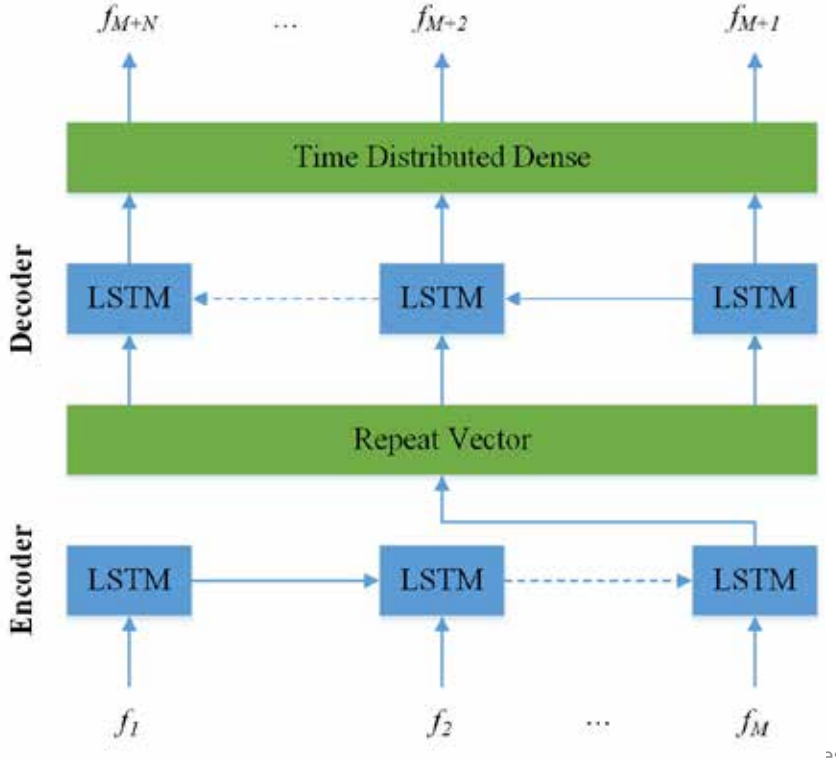


Fig. 1: LSTM encoder-decoder network for predicting yaw and pitch angles. f_1 to f_M are M input frames' features (yaw or pitch angles) and f_{M+1} to f_{M+N} are N output frames' features (yaw or pitch angles).

LSTM-based Viewpoint Prediction for Multi-Quality Tiled Video Coding in Virtual Reality Streaming

- Prediction adjustment:
  - Although LSTM is a good model for sequence prediction, the prediction error increases significantly for long-term horizons.
  - We improve the results obtained by the LSTM model:
    - We assume that some users are on low-latency networks, and that they are viewing the same content.
    - Their viewpoint information is used to adjust our model output.
    - The adjustment is made based on circular mean and circular variance of these guide users' yaw and pitch angles.

LSTM-based Viewpoint Prediction for Multi-Quality Tiled Video Coding in Virtual Reality Streaming

- Prediction adjustment:

$$Y_a = (1 - W) \times Y_p + W \times Y_g$$
$$W = \left(1 - V_g^{1/3}\right) \times (h / P_{max}),\ 0 \le V_g \le 1,\ 0 \le h \le P_{max}$$

Y_p: predicted yaw angle from LSTM model
Y_g: avg yaw angle of the guide users
Y_a: adjusted yaw angle of a target user
V_g: circular variance of guide users' viewpoints
W: confidence factor, between 0 and 1, of the guide users' information
h: time horizon (frame in the future we are predicting the viewpoint for)
P_{max}: maximum size of the prediction window

LSTM-based Viewpoint Prediction for Multi-Quality Tiled Video Coding in Virtual Reality Streaming

- Tiles quality assignment:
  - Non-tiled streaming reference at b Mbps.
  - Tile–based streaming with $m \times n$ tiles using
    - $b/(m \times n)$ Mbps for high quality tiles,
    - $b/(2 \times m \times n)$ Mbps for medium quality tiles
    - $b/(8 \times m \times n)$ Mbps for low quality tiles.

LSTM-based Viewpoint Prediction for Multi-Quality Tiled Video Coding in Virtual Reality Streaming

- Experimental conditions:
  - The dataset includes 48 users watching 9 videos.
  - For each video, one minute of the content is selected.
  - To train our model and to make predictions, we use the sine and cosine of yaw and pitch angles captured from each video and user.
  - Input sliding window and output prediction window are set to 20 and 120 frames respectively (M=20, N=120) and P_{max} is set to 200.
  - Grid of 12 x 12 tiles is applied to each video frame.
  - The videos are sampled at every 30 ms which means we are using a window of 600 ms to predict all the frames in the next 3.6 s.

LSTM-based Viewpoint Prediction for Multi-Quality Tiled Video Coding in Virtual Reality Streaming

- Results:

TABLE I: Prediction RMSE error and high- and medium-quality viewport for long-term horizon (3.6 s)

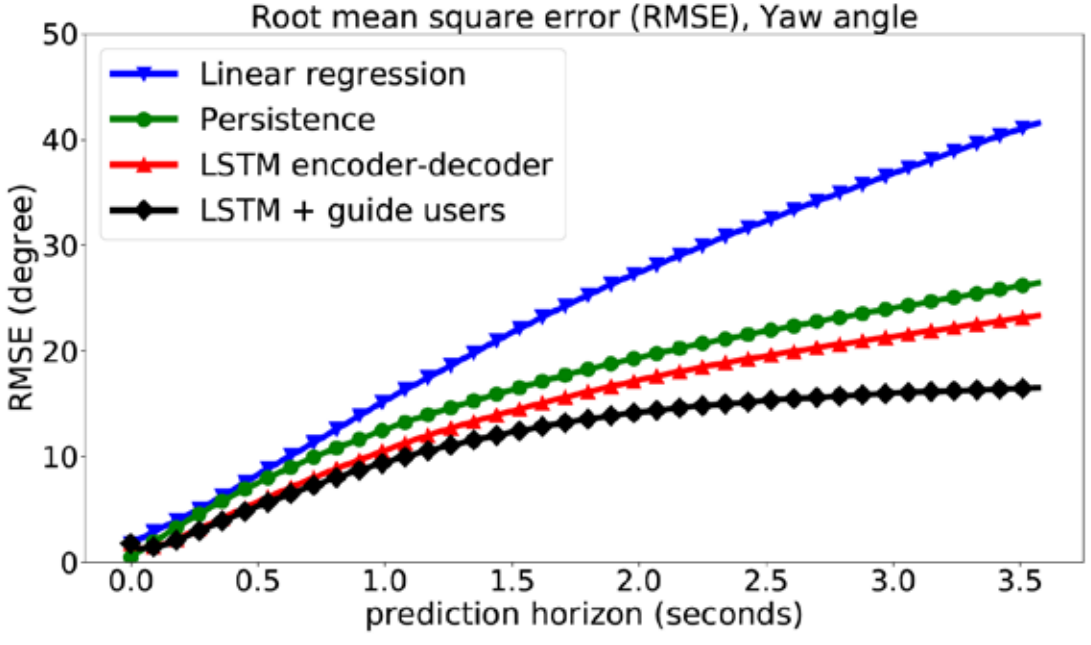
Method	Yaw RMSE (°)	Pitch RMSE (°)	HQV(%)	HQV + MQV(%)
Linear regression	41.6	13.1	70.2	93.8
Persistance	26.4	4.2	84.4	98.6
LSTM enc.-dec.	23.4	4.0	87.6	98.9
LSTM + guide users	16.5	3.6	95.9	99.7

Table I shows the error for both yaw and pitch and high-quality viewport (HQV) and the combination of HQV and medium-quality viewport (MQV). HQV and MQV show the overlap (in percentage) between the viewport seen by the user and the high-quality and medium-quality tiles, respectively.

LSTM-based Viewpoint Prediction for Multi-Quality Tiled Video Coding in Virtual Reality Streaming

- Results:

Prediction error based on RMSE for the entire prediction window.



We achieved 61% average bitrate reduction on average while maintaining excellent quality.

CU Size Decision for Low Complexity HEVC Intra Coding based on Deep Reinforcement Learning [R2]

- In video compression, coding units (CU) size determination / mode decision is very complex
  - An exhaustive search checks all possible modes and CU sizes to find the best coding parameters.
  - Some heuristics permit to reduce complexity at the cost of reduced quality/compression.
- We propose a novel CU size decision method based on deep reinforcement learning and active feature acquisition to reduce HEVC intra coding computational complexity (encoding time).
- The proposed method carries out early splitting and early splitting termination by considering the encoder and CU as an agent-environment system.
  - Through early splitting, the method precludes the need for rate-distortion optimization at the current level.
  - Through early splitting termination, it disposes of the lower level computations.
  - The proposed method provides a very fast encoder with a small quality penalty.
- Experimental results show that it achieves a **51.3% encoding time reduction** on average with a small quality loss of 0.041 dB for the BD-PSNR, when we compare our method to the HEVC test model.

[R2] M. Jamali, S. Coulombe and H. Sadreazami, "CU Size Decision for Low Complexity HEVC Intra Coding based on Deep Reinforcement Learning," 2020 IEEE 63rd International Midwest Symposium on Circuits and Systems (MWSCAS), Springfield, MA, USA, 2020, pp. 586-591, doi: 10.1109/MWSCAS48704.2020.9184456.

CU Size Decision for Low Complexity HEVC Intra Coding based on Deep Reinforcement Learning

- We deploy a batch-mode reinforcement learning (RL), based on fitted-Q iteration (FQI), to find a policy for CU size decision in the context of intra HEVC coding

Fig. 1: Example of frame splitting into CUs (red) and PUs (green) based on the HEVC quadtree structure, *RaceHorses*.

Algorithm 1 Encoder training phase (learning agent)

Input: $\mathcal{F} = \{(s, a, c, s')_l | l = 1, ..., n\}, A, N_A$

Output: $\hat{Q}_a^N$ as an approximation of the value function Q_a^N for all action $a \in A$

$k = 0$

initialization: set $\hat{Q}_{a_m}^0$ to zero everywhere on s, $m = 1$ to N_A

repeat

  for $m = 1$ to N_A **do**

    $\mathcal{F}_{a_m} = \{(s, a, c, s') \in \mathcal{F} | a = a_m\}$

    $\mathcal{T}_{a_m} = \{(i_l, o_l), l = 1, ..., n_{a_m}\}$ where:

    $i_l = s_l, o_l = c_l + \gamma \min_{a' \in A_{s'_l}} \hat{Q}_{a'}^k(s'_l)$

    Function approximation based on NN

    $\hat{Q}_{a_m}^{k+1} \leftarrow \text{NN}(\mathcal{T}_{a_m})$

  end for

  $k = k + 1$

until $k = N$

return $\hat{Q}_{a_m}^N$, $m = 1$ to N_A

CU Size Decision for Low Complexity HEVC Intra Coding based on Deep Reinforcement Learning

TABLE I: Experimental results while implementing RL-based CU size decision compared to HM

Class	Video sequence	TR (%)	BD-Rate (%)	BD-PSNR (dB)
A (2560×1600)	Traffic	-54.5	1.21	-0.059
	PeopleOnStreet	-54.4	1.25	-0.060
	Nebuta	-52.9	0.21	-0.018
	SteamLocomotive	-52.8	0.22	-0.019
B (1920×1080)	Cactus	-49.6	0.85	-0.029
	Kimono	-59.4	1.68	-0.056
	ParkScene	-50.2	0.76	-0.032
	BasketballDrive	-57.1	1.33	-0.033
	BQTerrace	-46.9	0.42	-0.028
C (832×480)	BQMall	-51.8	0.61	-0.036
	PartyScene	-37.6	0.27	-0.021
	RaceHorsesC	-48.5	0.52	-0.038
	BasketballDrill	-50.3	0.90	-0.039
D (416×240)	RaceHorses	-43.6	0.65	-0.049
	BasketballPass	-50.3	0.65	-0.042
	BlowingBubbles	-42.9	0.27	-0.012
	BQSquare	-42.7	0.19	-0.020
E (1280×720)	FourPeople	-61.1	1.61	-0.078
	Johnny	-58.3	1.73	-0.088
	KristenAndSara	-60.3	1.45	-0.063
Average (with training sequence)		**-51.3**	**0.84**	**-0.041**
Average (without training sequence)		**-51.7**	**0.85**	**-0.041**

CU Size Decision for Low Complexity HEVC Intra Coding based on Deep Reinforcement Learning

TABLE II: CU size decision methods comparison

	TR (%)	BD-Rate (%)
Proposed method	-51.3	0.84
[13]	-31.0	0.70
[6]	-55.5	1.01
[4]	-50.3	1.41

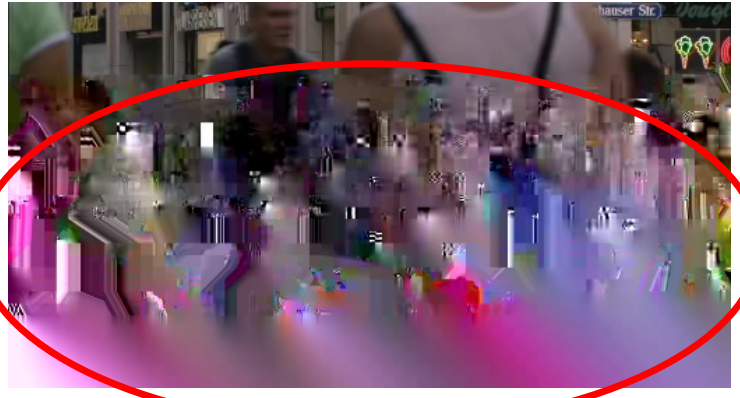
Discussion:

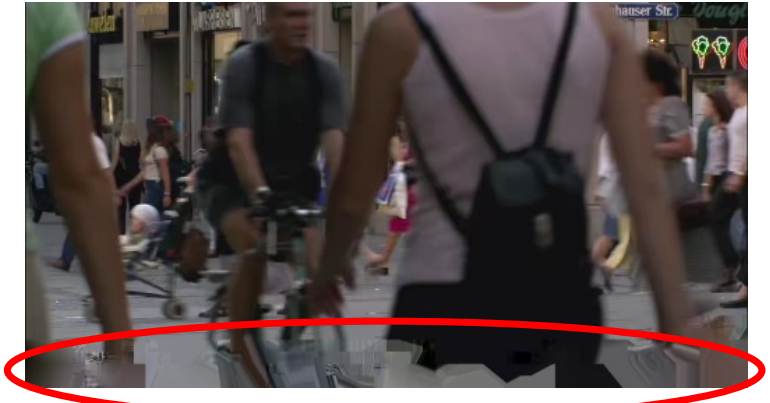
- AI can be used to find the best coding parameters for existing video compression methods/standards.
  - We can customize that to Immersive Video content.
- AI can also be used to develop new compression methods
  - E.g. auto-encoders for video and point couds compression

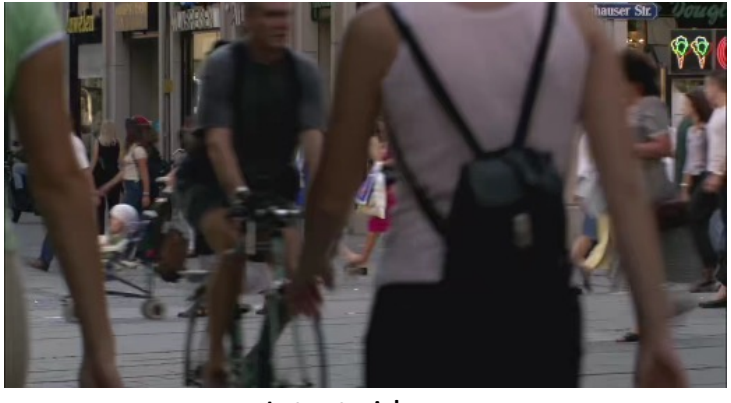
Optimization of list decoding of corrupted videos based on a CNN architecture [R3]

- We present an optimized list decoding solution for videos corrupted by transmission errors.
- It is based on image quality evaluation (without reference) using a neural network convolutional (CNN) that efficiently handles non-uniform distortions.
- After a list decoding process, we rate the quality of each candidate image generated in order to select the best one.
- When the transmission error occurs in an intra image, our architecture has a decision precision of more than 98% compared to 46% for the original pre-trained CNN architecture. For errors in an inter image, it's 79% against 33%.

[R3] Y. Zhang, S. Coulombe, F-X Coudoux, A. Trioux, P. Corlay, "Optimisation du décodage par liste de vidéos corrompues basée sur une architecture CNN," 22nd edition of COmpression et REprésentation des Signaux Audiovisuels (CORESA 2023), France, Lille, (June 2023) Conference Date: June 2023







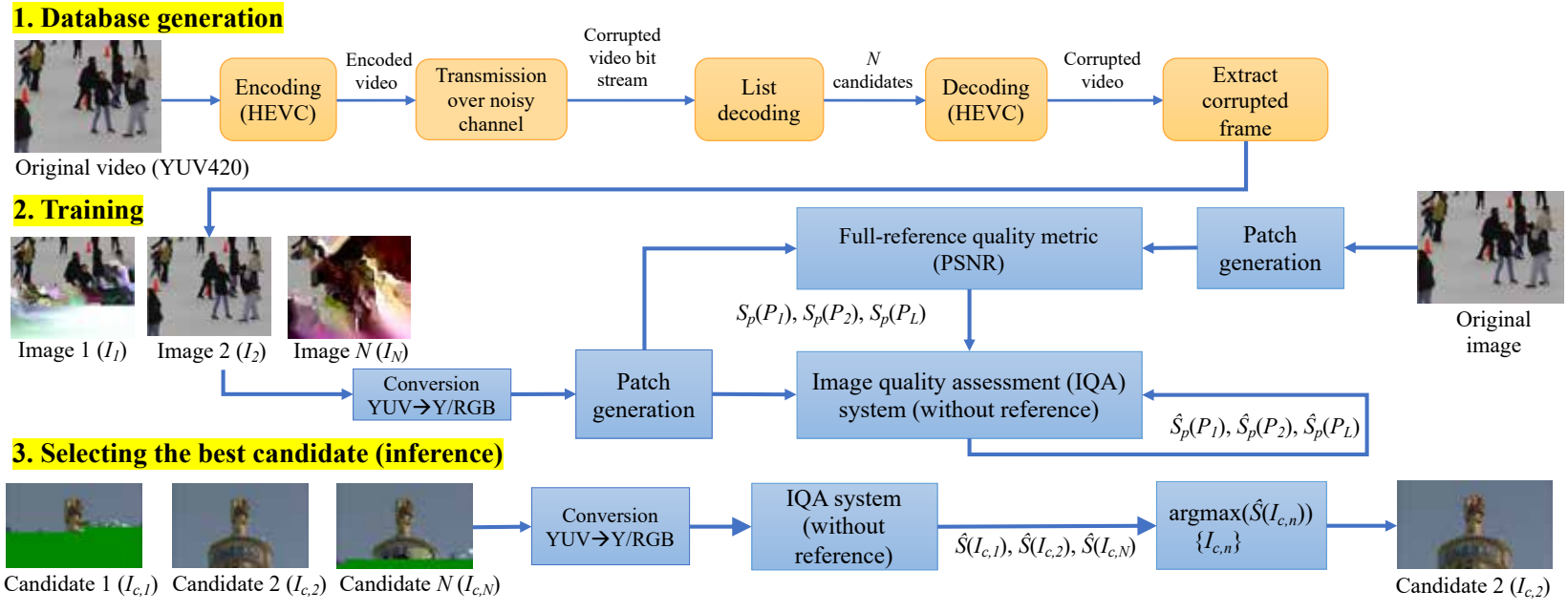
Intact video

Intact video

ÉTS Le génie pour l'industrie

ILLS International Laboratory on Learning Systems

Optimization of list decoding of corrupted videos based on a CNN architecture



Optimization of list decoding of corrupted videos based on a CNN architecture

Contributions:

1. A new quality assessment method based on a popular CNN, but improved in several aspects, including normalization and measurement of local quality, operating by patch, to support non-uniform distortions in images.
   - E.g. we fixed some issues like ambiguity with uniform patches.
   - Can be used with other CNN architectures.
2. A new database of encoded videos using the High Efficiency Video Coding (HEVC) standard and to which we injected errors transmission.
   - This leads to images with non-uniform spatially distributed artifacts on which our system can train.
3. A new *List Decoding Optimization Framework* able to select the video with the best quality visual among several candidates.

score: 0.4066

PSNR_YUV: 10.34

score: 0.6811

PSNR_YUV: 16.13

score: 0.7988

PSNR_YUV: 37.60

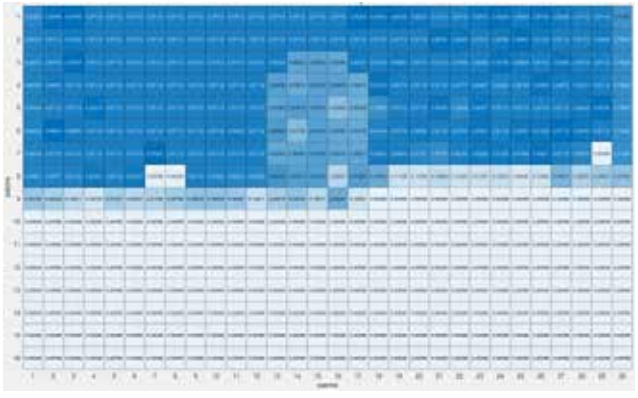
Choix de CNNIQA
score: 0.8033

PSNR_YUV: 43.38

Intact video

score: 0.4066

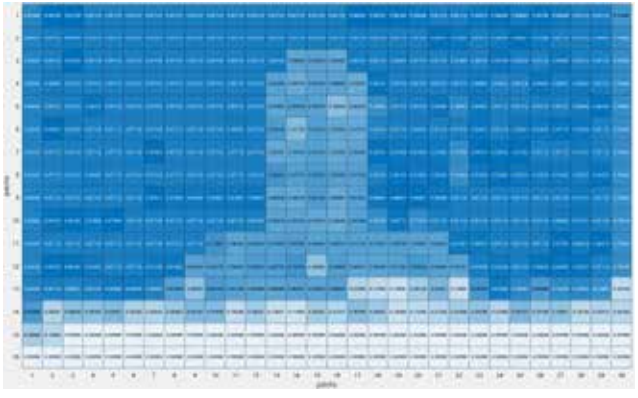
PSNR_YUV: 10.34



Légende du genre

score élevé

score faible

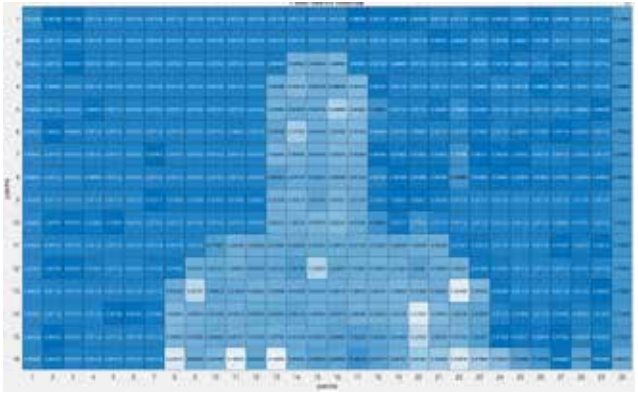


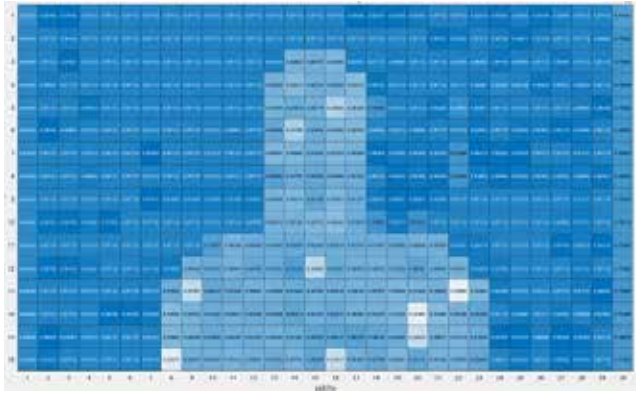
score: 0.6811

PSNR_YUV: 16.13

score: 0.7988

PSNR_YUV: 37.60





Choix de CNNIQA score: 0.8033

PSNR_YUV: 43.38

Intact video

Optimization of list decoding of corrupted videos based on a CNN architecture

Methods	Accuracy	$\overline{S}_{\text{intact}}$ (dB)	$\overline{S}_{\text{system}}$ (dB)	$\overline{S}_{\text{diff}}$ (dB)
CNN_Y_G pre-trained [7]	45.6%	39.18	28.69	10.49
CNN_Y proposed	93.0%		38.39	0.79
CNN_RGB proposed	96.5%		38.88	**0.30**
CNN_Y_NL proposed	**98.2%**		38.60	0.58
CNN_RGB_NL proposed	96.5%		38.88	**0.30**

TABLE 1 – *Performance on intra-coded images*

Methods	Accuracy	$\overline{S}_{\text{intact}}$ (dB)	$\overline{S}_{\text{system}}$ (dB)	$\overline{S}_{\text{diff}}$ (dB)
CNN_Y_G pre-trained [7]	33.3%	38.62	32.99	5.63
CNN_Y proposed	60.0%		30.19	8.43
CNN_RGB proposed	66.7%		36.49	2.13
CNN_Y_NL proposed	77.0%		36.55	2.07
CNN_RGB_NL proposed	**79.0%**		36.71	**1.91**

TABLE 2 – *Performance on inter-coded images*

$$\overline{S}_{\text{intact}} = \frac{1}{N}\sum_{n=0}^{N-1} \text{PSNR}_{\text{YUV}}(I_{\text{original},n}, I_{\text{intact},n}),$$

$$\overline{S}_{\text{system}} = \frac{1}{N}\sum_{n=0}^{N-1} \text{PSNR}_{\text{YUV}}(I_{\text{original},n}, I_{\text{system},n}),$$

$$I_{\text{system}} = \underset{\{I_{c,i},\ 0\le i<K\}}{\arg\max}\ \hat{S}(I_{c,i}),\quad \overline{S}_{\text{diff}} = \left|\overline{S}_{\text{intact}} - \overline{S}_{\text{system}}\right|$$

Future work:

- Improve the architecture to take into account correlation between patches.
- Apply to Immersive Video content

No-Reference Video Quality Assessment Using Distortion Learning and Temporal Attention [R4]

- Most NR-VQA models in prior art have been proposed for assessing a specific category of distortion, such as authentic distortions or traditional distortions.
- We propose a novel deep learning method for NR-VQA based on multi-task learning where the distortion of individual frames in a video and the overall quality of the video are predicted by a single neural network.
- Our method outperforms the state-of-the-art on traditional distortion databases such as LIVE VQA and CSIQ video, while also delivering competitive performance on databases containing authentic distortions such as KoNViD-1k, LIVE-Qualcomm and CVD2014

[R4] K. Kossi, S. Coulombe, C. Desrosiers and G. Gagnon, "No-Reference Video Quality Assessment Using Distortion Learning and Temporal Attention," in , vol. 10, pp. 41010-41022, 2022, doi: 10.1109/ACCESS.2022.3167446.

No-Reference Video Quality Assessment Using Distortion Learning and Temporal Attention

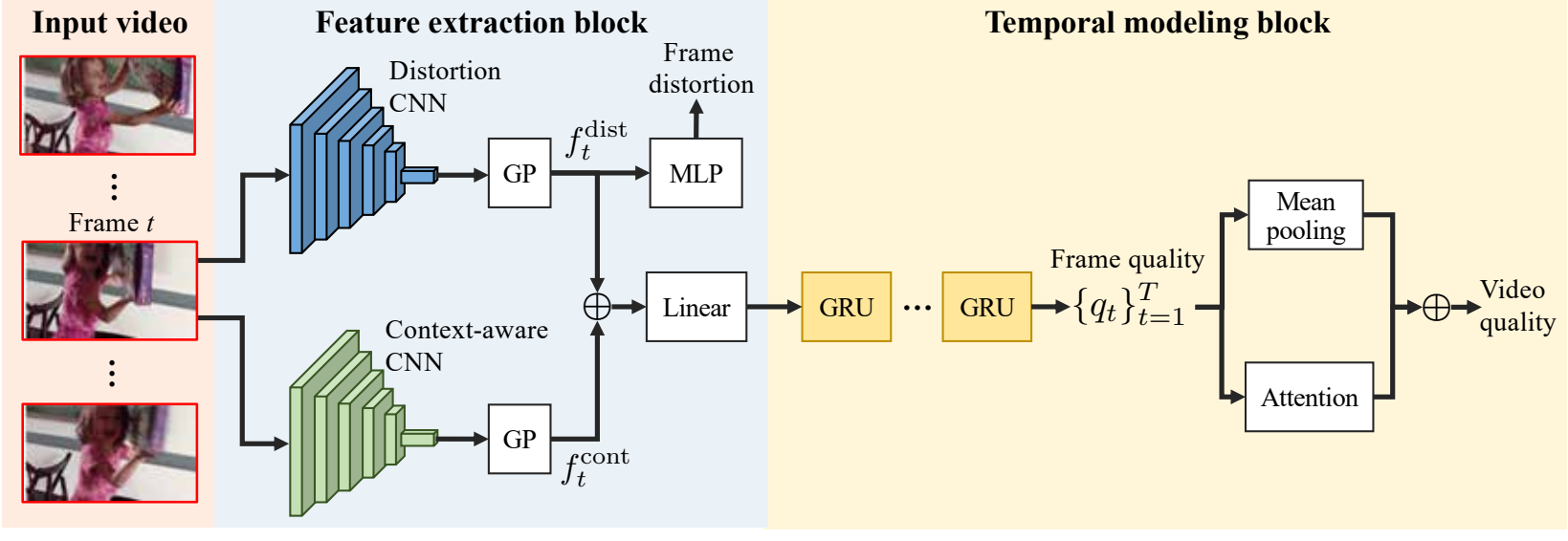


TABLE 1. Performance results on in-capture distortion databases. In each column, the best, and second-best values are respectively marked in boldface, and underlined. Note that * are performances taken from paper [38] and † from the methods' original papers. Other results were reproduced using the authors' code.

Method	Overall Performance			CVD2014		
	SROCC ↑	PLCC ↑	RMSE ↓	SROCC ↑	PLCC ↑	RMSE ↓
NIQE [11]*	0.39 (± 0.07)	0.40 (± 0.06)	4.24 (± 1.88)	0.58 (± 0.10)	0.61 (± 0.09)	17.1 (± 1.5)
BRISQUE [15]*	0.57 (± 0.06)	0.58 (± 0.06)	4.10 (± 0.45)	0.63 (± 0.10)	0.64 (± 0.10)	16.9 (± 2.2)
V-BLIINDS [32]*	0.62 (± 0.06)	0.61 (± 0.60)	3.72 (± 0.46)	0.70 (± 0.09)	0.71 (± 0.09)	15.2 (± 2.2)
HIGRADE [31]*	0.73 (± 0.04)	0.72 (± 0.04)	3.44 (± 0.37)	0.74 (± 0.06)	0.76 (± 0.06)	14.2 (± 1.5)
FRIQUEE [30]*	0.75 (± 0.04)	0.76 (± 0.04)	2.99 (± 0.29)	0.82 (± 0.05)	0.83 (± 0.04)	12.0 (± 1.2)
TLVQM [38]*	0.79 (± 0.03)	0.79 (± 0.03)	2.81 (± 0.33)	0.83 (± 0.04)	0.85 (± 0.04)	11.3 (± 1.3)
VSFA [13]	0.81 (± 0.03)	0.82 (± 0.03)	2.70 (± 0.03)	0.88 (± 0.03)	0.88 (± 0.03)	10.3 (± 1.2)
CNN-TLVQM [23]	**0.82** (± 0.03)	**0.83** (± 0.02)	**2.55** (± 0.03)	0.86 (± 0.04)	0.88 (± 0.03)	10.3 (± 1.1)
Proposed	0.81 (± 0.03)	0.82 (± 0.03)	2.56 (± 0.03)	**0.89** (± 0.03)	**0.90** (± 0.03)	**9.4** (± 1.4)

Method	KoNViD-1k			LIVE-Qualcomm		
	SROCC ↑	PLCC ↑	RMSE ↓	SROCC ↑	PLCC ↑	RMSE ↓
NIQE [11]*	0.34 (± 0.05)	0.34 (± 0.05)	0.61 (± 0.03)	0.46 (± 0.13)	0.48 (± 0.12)	10.7 (± 1.3)
BRISQUE [15]*	0.56 (± 0.05)	0.57 (± 0.04)	0.52 (± 0.02)	0.55 (± 0.10)	0.54 (± 0.10)	10.3 (± 0.9)
V-BLIINDS [32]*	0.61 (± 0.04)	0.58 (± 0.05)	0.53 (± 0.03)	0.60 (± 0.10)	0.67 (± 0.09)	9.2 (± 1.0)
HIGRADE [31]*	0.73 (± 0.03)	0.72 (± 0.03)	0.44 (± 0.02)	0.68 (± 0.08)	0.71 (± 0.08)	8.6 (± 1.1)
FRIQUEE [30]*	0.74 (± 0.03)	0.74 (± 0.03)	0.43 (± 0.02)	0.74 (± 0.07)	0.78 (± 0.06)	7.6 (± 0.8)
TLVQM [38]*	0.78 (± 0.02)	0.77 (± 0.02)	0.41 (± 0.02)	0.78 (± 0.07)	0.81 (± 0.06)	7.1 (± 1.0)
VSFA [13]	0.80 (± 0.02)	0.81 (± 0.02)	0.39 (± 0.02)	0.77 (± 0.07)	0.79 (± 0.06)	7.5 (± 0.9)
CNN-TLVQM [23]	**0.82** (± 0.02)	**0.82** (± 0.02)	**0.36** (± 0.02)	**0.81** (± 0.05)	**0.83** (± 0.03)	**6.73** (± 0.8)
RAPIQUE [24]†	0.80	**0.82**	**0.36**	–	–	–
PVQ [42]†	0.79	0.79	–	–	–	–
CoINVQ [25]†	0.76	0.77	0.41	–	–	–
Proposed	0.80 (± 0.02)	0.80 (± 0.02)	0.39 (± 0.02)	0.78 (± 0.07)	0.81 (± 0.06)	7.4 (± 0.9)

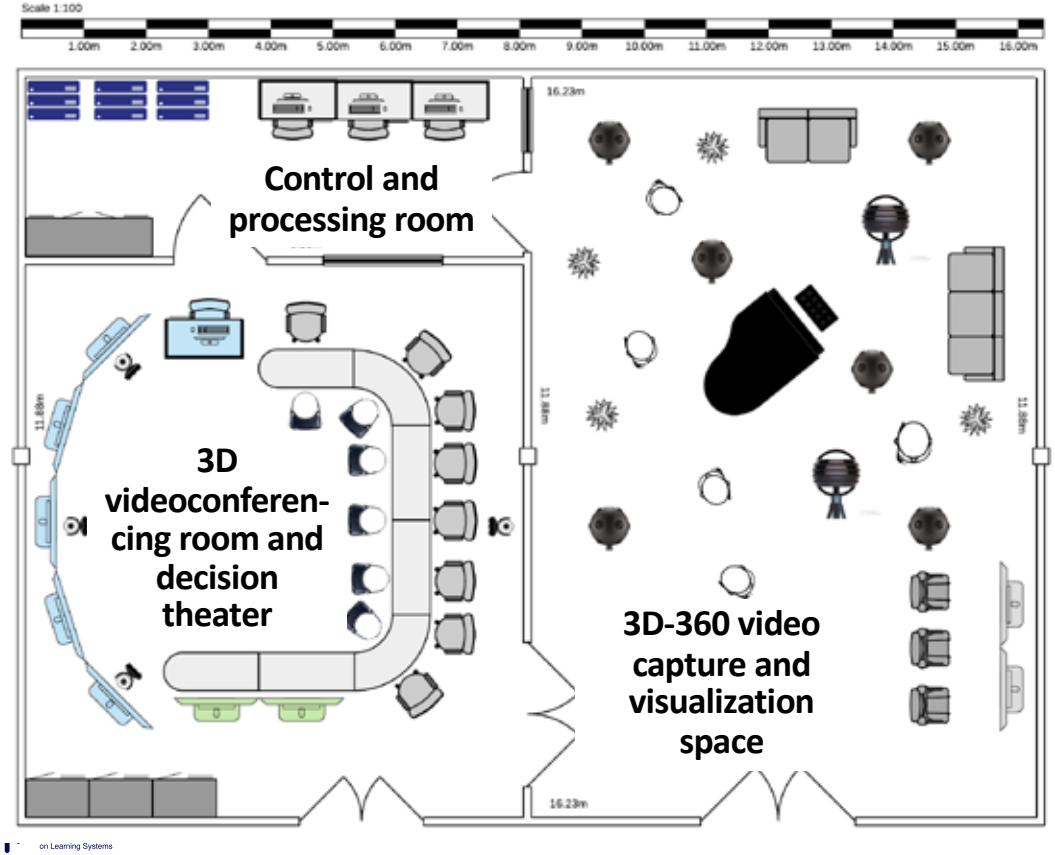
TABLE 2. Performance results on traditional distortion databases: CSIQ VIDEO and LIVE VQA. Note that * are performances taken from paper [18], [34] and † from the methods' original papers. Other results were reproduced using the authors' code.

Method	CSIQ VIDEO		LIVE VQA	
	SROCC↑	PLCC↑	SROCC↑	PLCC↑
NIQE [11]*	–	–	0.23	0.27
VIIDEO [12]*	-	-	0.62	0.65
V-BLIINDS [32]*	0.86	0.85	0.83	0.84
SACONVA [34]†	0.86	0.87	0.86	0.87
TLVQM [38]	0.74	0.74	0.60	0.62
VSFA [13]	0.76	0.74	0.73	0.75
V-MEON [36]†	0.82	0.82	–	–
CNN-TLVQM [23]	0.81	0.79	0.77	0.79
RAPIQUE [24]	0.76	0.77	0.66	0.72
Proposed	**0.91**	**0.91**	**0.89**	**0.90**

Future Work

- Improving our (AI) solutions for:
  - Viewport prediction
  - Content compression
    - 360-degree and 3D-360-degree video compression
  - Error concealment
- Developing new/improved AI solutions:
  - Content compression
    - Immersive video compression using MPEG Immersive Video standard (part of MPEG-I)
    - Point Clouds
  - Stitching
  - Depth estimation + View interpolation (6 DoF)
  - Visual quality assessment for Immersive Video
  - Etc.

Immersive video & decision theater lab (floor plan)



- Caméra UHD PTZ sur trepied ou au plafond
- Casque de réalité virtuelle (HMD)
- Écran autostéréoscopic 65" (sur chariot)
- Utilisateur avec casque de RV/RA
- Écran 70" (sur chariot)
- Table configurable sur roulettes
- Caméra 3D-360 (Z-Cam Pro, Vuze+, Insta360, Vantrix)
- Caméra Lightfield (Lytro ou Raytrix)
- Serveurs de calcul, équipement réseau et vidéo
- Cabinets de rangement
- Postes de travail avec ordinateurs
- Postes de travail avec ordinateur pour facilitateur

3D-360 video capture and visualization space

Pimax Vision 8K Plus

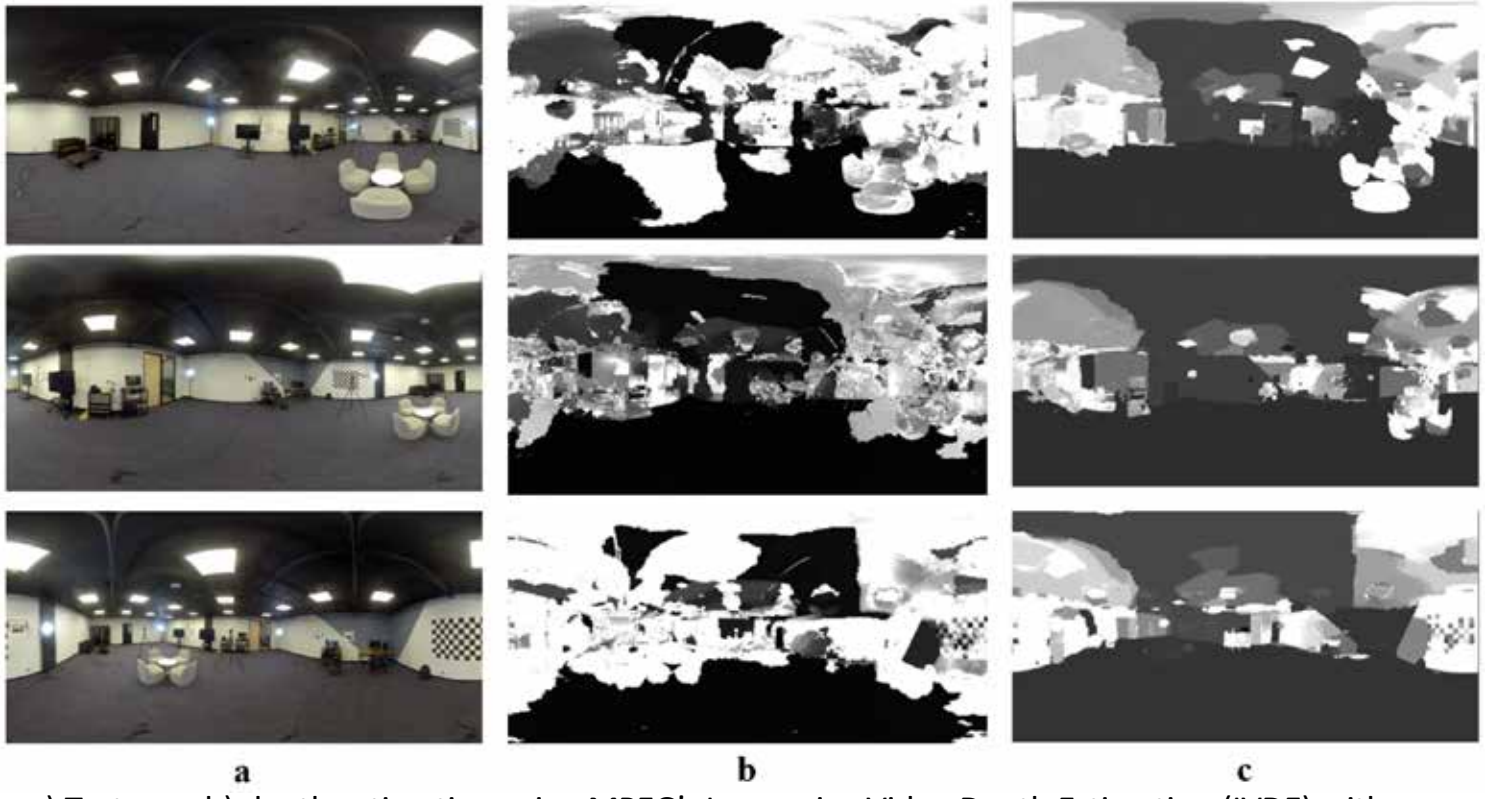
Future Work

- Depth estimation + View interpolation (6 DoF)





Depth estimation using multiple cameras



a) Textures, b) depth estimation using MPEG's Immersive Video Depth Estimation (IVDE) with automatic depth range, c) depth estimation using IVDE with improved depth range.

Stitching

Challenges:

- High visual quality
- Low complexity
  - Real-time

Conclusions

- Immersive video is a fast-growing technology with many applications
- It introduces many (interrelated) challenges:
  - Low latency / high bitrate / transmission errors
  - High computational complexity (compression)
  - Visual quality
- Already AI can help
  - Viewport prediction
  - Content compression
  - Error concealment
- Many other areas where AI can help…


Questions?
SUMMIT
NSERC
CRSNG
MITACS