DV2DM: A Learning-based Visible Difference Predictor for Videos

Anonymous Authors

***Abstract*—Video has emerged as a predominant medium in today's digital era. Efficient storage and transmission of video necessitate encoding and compression, yet excessive compression can significantly degrade quality. Visible Difference Predictor (VDP) metrics are pivotal in predicting and quantifying the visibility of image distortions, relying on models of the human visual system (HVS). While deep learning enables advanced image VDPs, video VDPs encounter greater complexities. In this work, we introduce the Deep Video Visible Difference Metric (DV2DM) and the ViLocVis dataset, the first dataset tailored for training deep video visible difference metric. The ViLocVis dataset, a hybrid of manually annotated and synthetic data, accounts for variables including viewing distance, display luminance and frame rate. Utilizing a calibrated spatio-temporal display model coupled with a customized U-ViT architecture, our approach captures spatio-temporal features effectively, demonstrating significant performance improvements. In addition, we provide several applications based on the DV2DM within the fields of machine learning and computer vision, showcasing its generalizability.**

***Index Terms*—DV2DM, Visible Difference Predictor, Video Quality Metric, Human Visual System**

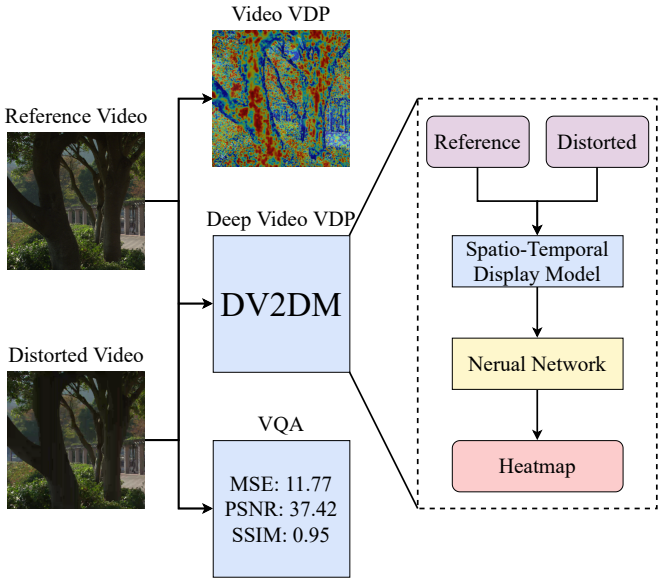

Fig. 1. The framework of our DV2DM, which consists a display model and a nerual network. Different from video quality assessment (VQA), visual difference predictor predicts the visual heatmap instead of a single value.

I. Introduction

VISIBLE Difference Predictor (VDP), intending to predict a map of pixel-wise visible difference from image or video pairs, plays an important role in giving effective suggestions on analyzing difference based on human visual system (HVS). With the development of communication technology, especially the rapid popularization of mobile Internet, the application of video in daily life is becoming more and more frequent. The urgent need for video compression means that better metrics need to be found. Such prediction holds significant potential within the domain of computer vision, such as image and video compression, reconstruction, rendering and the emerging technologies of virtual reality (VR) and augmented reality (AR).

Note that VDP is quite different with similar field of research, quality metrics, which aims to evaluate the quality of an image or video by a single value. On the one hand, VDP considers the human vision system factors hence it is a subjective metric. On the other hand, The VDP algorithm, in contrast, intends to returns a pixel-wise map instead of a single value, which providing with difference details on each location of the image [44], [45].

Most existing VDP work focuses on the image domain. However, with the proliferation of mobile Internet, video has become one of the most popular media today, making the need for video VDP increasingly urgent. For example, the transmission and storage of video almost always involves compression algorithms. In general, compression algorithms strive to reduce the storage space occupied by the video while losing as little picture information as possible. However, classical video compression algorithms often ignore the subjective perception of the viewer, which means over-compression and computational waste on the discernible areas, leading to an impaired viewing experience [13].

However, compared to VDP for images, video VDP faces more challenges, such as time dimension and dynamic content. Due to the temporal continuity between frames, the distortion may persist for several frames, affecting the overall perception of the viewer [28]. In addition, it is difficult to cover every frame of the video with annotation information, which makes predicting visible differences in a dynamic scene more complex than in a static image. Different video frame rates can affect the visibility of distortion. For example, a either too high or too low frame rate might spoil the visibility of differences, for sake that high frame rate may results into less perceptible details and too slow frame rate may shorten the relative window of visibility [40].

In this work, we explore Deep Video Visible Difference Metric (DV2DM), a method for evaluating and predicting visual differences or distortions in video content which can be noticed by the observer, quantifying visual differences

and simulating human visual response [42]. In addition, in order to train DV2DM, we collect the open-source video datasets and produce the ViLocVis dataset, the first dataset tailored for training deep video visible difference metric. The ViLocVis dataset is composed of manually annotated data and synthetic data, which consists of more than 1,000 video pairs and provides video annotations under various combinations of viewing conditions.

This work provides a new study in the field of machine learning and computer vision, which is significant for measuring and enhancing video quality for the use of human viewers' perceptual information, especially for this era of video media. The ViLocVis dataset addresses the issue of lack of manually annotated data to some extent within this research field, paving the way for further exploration. Moreover, we extend the application of DV2DM to various real-life scenarios and other research areas such as machine learning and computer vision. Specifically, our main contributions are as follows:

- We provide the subsequent research on video VDP with a novel dataset ViLocVis of over 1000 video paris, which includes annotation data from 10 volunteers across 11 open-source video datasets under various viewing conditions, along with synthetic data containing 12 different viewing conditions.
- We propose an effective method to apply deep learning to the video VDP and achieve the state-of-the-art performance, which is based on a U-ViT backbone tailored for video VDP. The model architecture utilizes a Siamese U-shaped network architecture with two encoding branches and only one decoding branch.
- We consider multiple environmental factors, such as viewing distance, display luminance and frame rate, which enables the algorithm to better capture and model the relationships between these variables.
- We provide examples of applications based on DV2DM. For instance, content-adaptive watermarking, visually lossless video compression, invisible adversarial attacks and video super-resolution metric.

II. Related Work

A. Human Visual System and Quality Metrics

1) Human Visual System: The Human Visual System (HVS) refers to the biological system facilitating visual perception, encompassing the eyes, neural pathways, and parts of the brain that process visual data [25]. The study of HVS is important for designing efficient algorithms from the perspective of human eye perception. For example, the HVS is leveraged to design efficient video and image compression algorithms, where information less perceived by the human eye might be deprioritized or omitted during compression [22], [35].

2) Image Quality Metrics: The measurement for quality of an image is quite a mature field of research. Fundamental approaches (e.g. MAE, MSE, PSNR) simply measure the average difference in whole. SSIM [39], MS-SSIM [38], and FSIM [47] further account for the contextual information and structural similarity of image pairs. Machine-learning-based method [5] treats the quality metric problem as a classification process, where the quality is classified into a five-level scale based on the Support Vector Machine, whereas suffering from the subjective definition on quality levels. [48] argues that human perception on similarity depends on high-order structure and is context-dependence, hence introduced a CNN-based method, which is demonstrated to outperform substantially.

Recent advancements in display technologies highlight limitations in existing image quality metrics which assume consistent dynamic ranges between reference and test images. Addressing this, [3] introduced a novel metric that can compare images with varying dynamic ranges, leveraging insights from the human visual system to detect and classify visible structural changes in images. The utility of this metric was further demonstrated in evaluating tone mapping operators and diverse display characteristics.

3) Video Quality Metrics: Video quality is evaluated in different ways from image, due to differences in HVS's handling dynamic and static content. For example, high-frequency noise can be clearly shown in image, whereas less noticeable in high-frame-rate video due to fusion of successive frames [19].

Video Quality Assessment (VQA) techniques are bifurcated into no-reference and full-reference metrics. No-reference metrics evaluate videos relying solely on the test video itself [31] while full-reference metrics compare distortions in the test video against the reference video. Early studies, notably [41] used basic statistical methods to model discrete cosine transform (DCT) quantization noise visibility in digital video quality metrics. [43] discusses the evolution of video quality metrics towards hybrid models blending objective and subjective elements. Machine learning advancements leverage accuracy and consistency [23], [30]. Deep learning further advances with end-to-end neural network [1], [12], [14]. [12] further takes spatio-temporal perception CNN into account. For HDR videos, [24] introduces a calibrated method focusing on noticeable distortions through spatial segmentation and temporal pooling.

B. Visible Difference Predictor

1) VDP: While VQA provides overall video quality assessment through scores, VDP offers finer pixel-level predictions of visible differences in images or videos, which emulates human visual perception to predict the noticeability of visual distortions. It can be traced back to [9] and subsequent research tailors white-box visibility metrics to assess the possibility for human to detect contrasts between image pairs.

The Spatial-CIELAB metric [49] focuses on the color reproduction errors of digital images. It extends the traditional CIELAB [10], which is originally designed for matching large uniform colored areas, to general chromatic images, by tailoring color transformation and spatial filters according to psychophysical experiments on the HVS. Note that, the spatial kernels relies on a Contrast Sensitivity Function (CSF) taking the exponential form, which is over-simplified and cannot generalize well to the more complex cases of images.

Subsequent to early VDP models [9], [18], [27], which incorporated complex low-level vision models but were limited by data scarcity, hindering extensive machine learning application. These models relied on detailed HVS models with minimal trainable parameters, restricting efficacy in complex scenarios. [44] mitigates this by introducing a new dataset for image distortion visibility, enabling machine learning advancements. [45] expands this by incorporating viewing condition factors such as luminance and viewer distance.

With the shift to High Dynamic Range (HDR) imaging, [17] enhances VDP to accurately predict visible differences in HDR scenarios, considering HDR display contrast ratios and real scene adaptation conditions. This approach, calibrated with high-end HDR displays, also accounts for optical light scattering and local adaptation. Further advancements in HDR-VDP are noted [18], [24].

The trade-off between accuracy and computational efficiency in SDR and HDR metrics remains a challenge. Metrics like HDR-VDP2.2 [24] emulate the HVS well but are computationally demanding. Simpler metrics like PSNR or MSE often miss critical HVS elements. To balance efficiency and accuracy, [4] introduces NoR-VDPNet++, a CNN-based metric optimizing the precision of HDR-VDP2.2.

2) Video VDP: In video contexts, visibility difference prediction necessitates additional considerations like spatio-temporal characteristics and spatial error signals [12]. Despite its significance, research on video VDP remains scarce. [19] pioneers by introducing the first video VDP tailored for foveated displays, such as VR or head-mounted displays, and involves calibration of psychophysical models and the HVS.

III. Data Collection

To collect responses of the human visual system (HVS) to distorted videos under various luminance and viewing distance, we produce a dataset named ViLocVis, which comprises of several reference-distorted video pairs and corresponding marking videos. The dataset can be divided into two parts: manually annotated data and synthetic data.

TABLE I
Datasets Details

Dataset	Videos	Resolution	FPS
HEVC-B [36]	5	1920x1080	24, 50, 60
HEVC-C [36]	4	832x480	30, 50, 60
HEVC-D [36]	4	416x240	30, 50, 60
HEVC-E [36]	6	1280x720	60
HEVC-F [36]	4	1280x720, 1024x768, 832x480	20, 30, 50
LIVE-APV [32], [33]	45	3840x2160	25, 30
LIVE-SJTU [21]	14	1920x1080	24, 25, 30
MCL-JCV [37]	30	1920x1080	24, 25, 30
MCML [7]	10	3840x2160	30
SJTU4K [34]	15	3840x2160	30
UVG [20]	16	3840x2160	50, 120

The reference videos are from open-source datasets, including the HEVC Standard Test Sequences (Class B, C, D, E, and F) [36], LIVE-APV [32], [33], LIVE-SJTU [21], MCL-JCV [37], MCML [7], SJTU4K [34] and the Ultra Video Group (UVG) dataset [20]. Table I provides a summary of the datasets. These videos cover a wide range of resolutions and frame rates in common scenarios.

A. Data Preparation

Before the annotation process, considering that higher video resolutions result in increased storage and computational costs, the videos with a resolution with 3840x2160 in the original datasets are downsampled to 1920x1080. Additionally, to facilitate the annotation process by enabling annotators to simultaneously view reference and distorted videos, thereby enhancing the ability to discern and annotate distortions in the videos, we crop the videos into a square format to align with the resolution ratio of the annotation display monitors.

Since the most common distortion in video is caused by compression, we employ compression method to generate distorted videos. Considering the popularity of compression encoders, we select the widely used H.264 and H.265. During the compression process, we appropriately adjust the compression parameters to ensure that most distortions in the compressed videos are perceptible.

It is note-worthy that there are also other common types of distortion, such as noise, blur and watermark. In addition, new distortions introduced by neural networks are also worth attention with respect to the emerging of learning-based compression and reconstruction algorithm [2], [6], [11], [16]. However, as an initial work, we carefully start with the classical compression algorithm and stress out the importance of our first step into locally annotated videos.

B. Manually Annotated Data

In order to understand human's visibility on distortion in videos, we conduct a set of experiments to collect a dataset from participants. Specifically, we recruit 10 subjects to perform video annotation in a unified experimental environment. In Figure 2, our annotation software is shown. The left and right image regions show the reference video and the distorted video respectively in the pause state.

After the participant starts the annotation, the reference video and the distorted video will playing at the same time. During this process, the participant is required to perceive the distortion is the video while the video is playing, and memorize the location and the frame id. It is worth mentioning that we ask the participant not to pause during the video playback to perceive the distortion in the still image. At the end of the video playback, both the left and right image regions will display the reference video, at which time, the participant can pause the video and rewind the video back to the specified frame for annotation. In this way, the distortion labeled by the participant is not the distortion perceived in the still image, but the distortion perceived in the dynamic process.

The observers view the display at all combinations of three distance levels (40 cm, 65 cm and 80 cm, which correspond to pixel per degree of 30, 50 and 60, respectively) and three screen luminance levels (110 $\mathrm{cd/m^2}$, 200 $\mathrm{cd/m^2}$, and 300 $\mathrm{cd/m^2}$).

Fig. 2. The user interface of our labelling tool in the pause state, where the layout is: A) the reference video, B) the distorted video, C) the progress bar, D) Current frame and total frames, and E) Number of frames have been saved.

Fig. 3. The experiment environment, where the left photo shows the physical environment for subjects to label the videos, and the right photo shows the same environment at the actual ambient lighting conditions.

However, allowing subjects to annotate the distortion videos frame by frame from memory poses a series of challenges, such as the inability to remember the distortion of each frame in a single viewing of the video, and multiple viewings of the video leading to subject fatigue and adaptation to the distortion, which affects the quality of the labeling. To avoid frame-by-frame annotation while obtaining annotated data with high fidelity, we utilize the Kernel Density Estimation (KDE) method and the optical flow method to achieve the spatio-temporal sparse annotation of videos, where subjects only need to annotate a few frames.

1) KDE: The Figure 4 shows the different levels of distortion and labeling that subjects perceived for the same video under different viewing conditions. It is obvious that the subjects label the areas where they could perceive distortion as 1 and the other areas as 0. On the one hand, due to the labeling principle, the subjects cannot label the probability of perceiving distortion, but only whether or not they could perceive distortion. On the other hand, such a sparse annotated data is not conducive to the training process of the neural network, as the neural network will quickly fall into local optima during training, outputting a heatmap entirely filled with 0. Therefore, it is necessary to employ the KDE method to transform the sparse annotated data of the annotated frames into dense annotated data, simultaneously achieving probability density modeling.

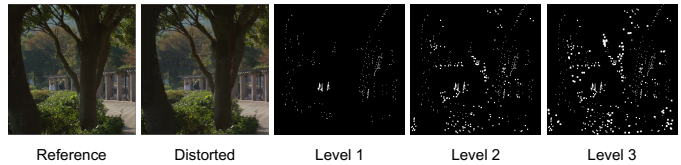

Fig. 4. Different levels of labelling: {"Level 1", "Level 2", "Level 3"} showcases different levels of labelling from coarse to fine, e.g. luminance from low to high, ceteris paribus.

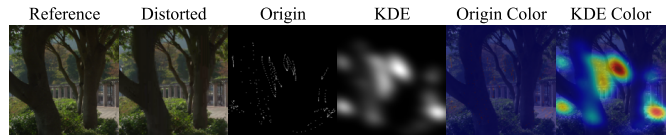

Fig. 5. The annotations before and after applying the KDE method.

The KDE method [53], [54] is a non-parametric estimation method for estimating the probability density function. The KDE method is commonly used to approximate the true distribution of data through smoothing data points, particularly when there is no prior knowledge about the data distribution. Suppose the data points are represented as $[x_1, x_2, \ldots, x_n]$, its density estimator is $\hat{f}_h(x) = \frac{1}{n}\sum_{i=1}^{n} K_h(x - x_i) = \frac{1}{n}\sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right)$, where K is the kernel function and h is the bandwidth.

Since the perception process of the human eye can be approximated as a 2d-Gaussian distribution, we model the annotation process in the labeled frames as the following problem: The data points are $[(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)]$ and the kernel function is the 2d-Gaussian kernel, the density estimation of any point (x, y) is

$$\begin{aligned}\hat{f}_{h_1,h_2}(x, y) &= \frac{1}{nh_1h_2}\sum_{i=1}^{n} K\left(\frac{x - x_i}{h_1}\right) K\left(\frac{y - y_i}{h_2}\right) \\ &= \frac{1}{nh_1h_2}\sum_{i=1}^{n} \frac{1}{\sqrt{2\pi}h_1} e^{-\frac{(x-x_i)}{2h_1^2}} \frac{1}{\sqrt{2\pi}h_2} e^{-\frac{(y-y_i)}{2h_2^2}}\end{aligned} \tag{1}$$

Therefore, through the KDE method, it is possible to achieve sparse annotation within labeled frames of videos. The comparison between annotations before and after applying the KDE method is shown in Figure 5. It's evident that the annotated data processed with KDE maintains semantic consistency with the original annotated data while becoming denser and more continuous on the spatial level.

2) Optical Flow: As the subjects only need to label several frames, we employ the optical flow estimation method to obtain the annotated data of other unlabeled frames. According to the optical flow method, there is $X(x+u, y+v, t+1) = X(x, y, t)$, where u and v are the optical flow. Additionally, due to the directionality of optical flow, we use u_f, v_f to denote the forward optical flow and u_g, v_g to denote the backward optical flow. Then we can define the functions f

and g as follows:

$$\begin{aligned} f(X(t)) &= X(t+1) \\ &= X(x+u_f(x,\,y,\,t),\,y+v_f(x,\,y,\,t),\,t) \\ g(X(t)) &= X(t-1) \\ &= X(x+u_g(x,\,y,\,t),\,y+v_g(x,\,y,\,t),\,t) \end{aligned} \tag{2}$$

Further, we can define $X(t+2) = f(X(t+1)) = f(f(X(t))) = f \circ f(X(t)) = f^2(X(t))$ and $X(t-2) = g(X(t-1)) = g(g(X(t))) = g \circ g(X(t)) = g^2(X(t))$.

It is note-worthy that due to errors inherent in optical flow method, the relationship $X(t) = f(X(t-1)) = g(X(t+1)$ does not universally hold. Moreover, the intervals between annotated frames may be substantial. To reduce the error of the optical flow method while modeling the long-range relationship, we introduce two confidence factors ω_f and ω_g to represent the confidence levels for forward and backward optical flow, respectively. Thus, we can get $X(t) = \dfrac{\omega_f f(X(t-1)) + \omega_g g(X(t+1))}{\omega_f + \omega_g}$. Furthermore, these confidence factors undergo exponential decay as they relate to distance, leading to the following formulation $X(t) = \dfrac{\omega_f^a f^a(X(t-a)) + \omega_g^b g^b(X(t+b))}{\omega_f^a + \omega_g^b}$.

Since distortions within videos often correlate with the content of the region and exhibit good temporal consistency, it can be assumed that the annotated data for distorted regions also maintains excellent temporal consistency. This implies that the optical flow calculated from reference video aligns with that derived from labeled video. Consequently, by computing the optical flow from the reference video and then applying it on labeled frames, the annotations for the entire video sequence can be obtained. Let X_{ref} and Y represent the reference video and labeled video separately, there is $X_{\text{ref}}(t+1) = f(X_{\text{ref}}(t))$, $Y(t+1) = f(Y(t))$.

Assuming that subjects only annotate frame $Y(t_1), Y(t_2), \ldots, Y(t_n)$, where $t_1 < t_2 < \cdots < t_n$, we can get the annotated data of the entire video as follows:

$$Y(t) = \begin{cases} Y(t) & \text{if } t \in \{t_1,\,t_2,\,\ldots,\,t_n\} \\ g^{t_1-t}(Y(t_1)) & \text{if } t < t_1 \\ f^{t-i_n}(Y(t_n)) & \text{if } t > t_n \\ \dfrac{\omega_f^a f^a(Y(t_m)) + \omega_g^b g^b(Y(t_{m+1}))}{\omega_f^a + \omega_g^b} & \text{if } t_m < t < t_{m+1} \end{cases} \tag{3}$$

where $a = t - t_m, b = t_{m+1} - t, m \in \{1,\,2,\,\ldots,\,n-1\}$. Therefore, we can achieve the temporal sparse annotation.

3) Order of KDE and Optical Flow: Although the sparse annotation can be achieved by KDE method and optical flow method, the order of applying KDE and optical flow method remains an issue.

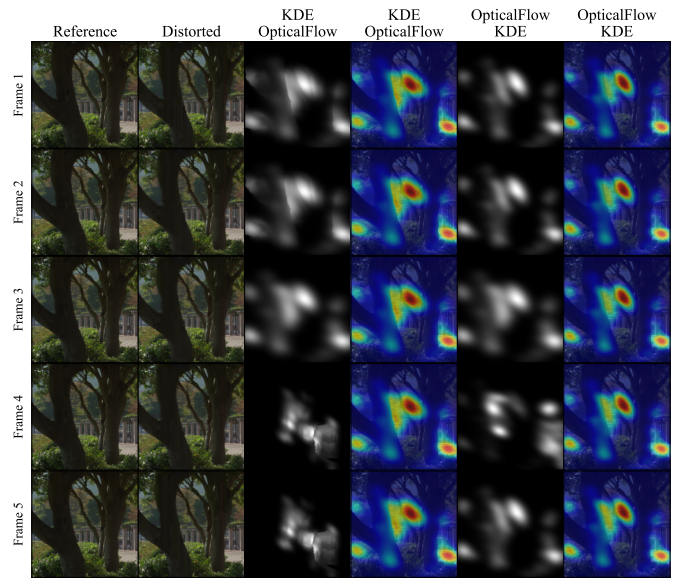

Fig. 6. The annotations under different orders of KDE and optical flow.

Firstly, if we apply the KDE method before the optical flow method, according to the Equation 1, we can get the data distribution of labeled frame is $Y(x,\,y,\,t) = \dfrac{1}{nh_1h_2}\sum_{i=1}^{n} K\left(\dfrac{x-x_i}{h_1}\right) K\left(\dfrac{y-y_i}{h_2}\right)$. In the case of considering only the forward optical flow u_f, v_f, we can get

$$\begin{aligned} Y_1(x,\,y,\,t+1) &= Y(x+u_f(x,\,y,\,t),\,y+v_f(x,\,y,\,t),\,t) \\ &= \frac{1}{nh_1h_2}\sum_{i=1}^{n} K\left(\frac{x+u_f(x,\,y,\,t)-x_i}{h_1}\right) \\ &\quad \cdot K\left(\frac{y+v_f(x,\,y,\,t))-y_i}{h_2}\right) \end{aligned} \tag{4}$$

Then, if we apply the KDE method after the optical flow method, suppose the data points in the t frame are $[(x_1,\,y_1),\,(x_2,\,y_2),\,\ldots,\,(x_n,\,y_n)]$, the data points obtained by the optical flow method in the $t-$th frame are $[(x_1 + u_f(x_1,\,y_1,\,t_1),\,y_1 + v_f(x_1,\,y_1,\,t_1)),\,(x_2 + u_f(x_2,\,y_2,\,t_2),\,y_2 + u_f(x_n,\,y_n,\,t_n)),\,\ldots,\,(x_n + u_f(x_n,\,y_n,\,t_n),\,y_n + v_f(x_n,\,y_n,\,t_n))]$. After applying KDE method in the $(t+1)-$th frame, we can get

$$\begin{aligned} Y_2(x,\,y,\,t+1) &= \frac{1}{nh_3h_4}\sum_{i=1}^{n} K\left(\frac{x-x_i-u_f(x_i,\,y_i,\,t_i)}{h_3}\right) \\ &\quad \cdot K\left(\frac{y-y_i-v_f(x_i,\,y_i,\,t_i)}{h_4}\right) \end{aligned} \tag{5}$$

By comparing the Equation 4 and the Equation 5, it is obvious that the Equation 4 takes into account the optical flow for each point while Equation 5 only considers the optical flow from the annotated points without utilizing the complete set of optical flow data.

As shown in the Figure 6, the annotated data obtained by both orders of KDE method and optical method appears

visually similar. However, employing the KDE method first, followed by the optical flow method, allows the annotation details to retain certain contours related to objects within the frame, such as the tree trunks. Conversely, applying the optical flow method before the KDE method results in generated annotated data that exhibits overall contour blurriness.

Hence, on the one hand, applying the KDE method first enables the effective utilization of all available optical flow data. Concurrently, it permits the annotated data to exhibit some alignment with the contours of relevant objects, thereby producing outcomes more consistent with human perception of video content.

On the other hand, due to the time complexity of the KDE method is $O(n^2)$, where n is the number of data points. Assuming that the subject labels t frames out of a video with T frames, with each frame containing n data points, the time complexity for first applying the KDE method is approximately $O(tn^2) + O_{\text{optical}}$, where O_{optical} denotes the time complexity for calculating optical flow. Furthermore, since the optical flow estimation method may not map coordinates onto integer coordinate points, it's often necessary to perform bilinear interpolation to distribute non-integer coordinate data across surrounding integer coordinate points. Hence, the time complexity for first employing the optical flow estimation method approximates to $O(Tn_{\text{optical}}^2) + O_{\text{optical}}$. It is clear that $T > t, n_{\text{optical}} > n$, the time complexity of applying optical flow method first far exceeds the reverse sequence. As a result, from both temporal efficiency and effectiveness perspectives, employing the KDE method prior to the optical flow estimation proves superior.

C. Synthetic Data

Due to the time-consuming manual annotation process and the limited diversity in luminance and viewing distance of the manually annotated data, coupled with the substantial need for semantically rich annotations during the training and testing process of the model, we choose to utilize the white-box video visibilty metric FovVideoVDP [19] to generate a large volume of synthetic data for the pre-training stage of the model.

To collect diverse luminance and viewing distance data, we set luminance levels to 10 cd/m^2, 110 cd/m^2, 200 cd/m^2, and configure viewing distance measured in angular resolution as 30, 40, 50, 60 pixel per degree (PPD). Therefore, we build 12 observation environments and applied FovVideoVDP to generate the heatmap across these environments.

However, the heatmaps generated by FovVideoVDP cannot be directly utilized. It is because its pixel values do not represent the probability of an observer perceiving distortion at the corresponding pixel. Instead, they are in just objectionable difference (JOD) units, which stands the level of distortion and do not align with the physical meaning of the manually annotated data. Therefore, we convert the JOD values to probabilities using the following equation:

$$P(x,y) = \frac{1}{\sqrt{2\pi}\sigma} \int_{-\infty}^{J(x,y)-\min(J)} e^{\frac{-t^2}{2\sigma^2}} \, \mathrm{d}t \tag{6}$$

where $P(x,y)$ is the probability of an observer perceiving distortion at pixel (x,y), $J(x,y)$ is the JOD value of the pixel, $\min(J)$ is the minimum JOD value in the heatmap, and σ is the standard deviation of the JOD units.

IV. Metric Architecture

A. Spatio-Temporal Display Model

The accuracy and effectiveness of Video Visible Difference Predictor (VDP) relies on accurately simulating the human visual system (HVS), where three core aspects are of most significance, namely, pixel per degree (PPD), brightness (Luminance) and frame rate (FPS) [45].

Pixel per degree (PPD) reflects the impact of image resolution on visual clarity. High PPD means higher image details, which are closer to the perception of the human eye in a natural viewing environment. Therefore, this physiological characteristic must be considered when VDP is simulated and evaluated to avoid unnecessary investment of resources beyond the range of human visual resolution [45]. Let PPD denote the Pixels Per Degree, a crucial metric in screen resolution analysis that quantifies the number of pixels visible per degree of visual angle. Its computation is determined by the equation 7:

$$PPD = \frac{R}{W\theta} \tag{7}$$

where R refers to the screen resolution, W represents the screen width, and θ corresponds to the field of view, an angular measure representing the angle subtended by the observer and the screen, which can be further defined by the viewing distance D:

$$\theta = 2 \cdot \arctan\left(\frac{W}{2D}\right) \tag{8}$$

Luminance is another key factor in the display model, which directly affects the perception and visual comfort of the image. The visibility of difference in human eyes can be dramatically distinct by different brightness levels, and the grasp of this dynamic range is crucial in VDP [18]. To account for the variation of luminance, we adopt a transform model by the following equation:

$$L' = L_{\text{black}} + (L_{\text{peak}} - L_{\text{black}}) \left(\frac{I}{MAX_I}\right)^{\gamma} \tag{9}$$

where I is the original pixel value by R-G-B channels, MAX_I is the maximal pixel value in the corresponding channel, L_{peak} is the peak luminance, L_{black} is the luminance when displaying pure black and $gamma$ is the gamma value, usually 2.2.

Frame rate (frame per second, FPS) is also important in video VDP, especially when evaluating video content. FPS affects the smoothness and visual continuity of video streams, while HVS has significant differences in perception of different frame rates, especially when dealing with fast-moving scenes [40]. In order to allow the model to respond to videos with different FPS, we use interpolated frames to align the FPS to the target. The formula is as follows:

$$n = \left\lfloor \frac{\mathrm{FPS}_{\text{target}}}{\mathrm{FPS}_{\text{current}}} + 0.5 \right\rfloor \tag{10}$$

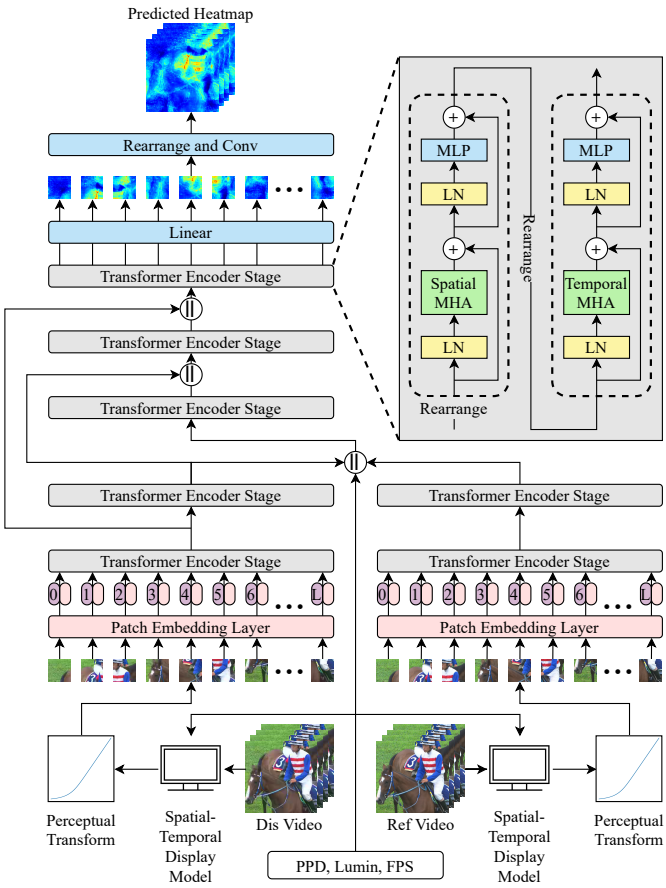

Fig. 7. The architecture of DV2DM, which is a customized U-ViT with two encoding branches and one decoding branch. Each transformer encoder stage contains two successive transformer encoder block, performing sptial attention and temporal attention separately.

where n is the number of repetitions per frame during frame interpolation, $\mathrm{FPS}_{\mathrm{current}}$ is the frame rate of the video, and $\mathrm{FPS}_{\mathrm{target}}$ is the target frame rate after frame interpolation.

B. U-ViT Architecture

The U-ViT architecture specifically tailored for Video Visible Difference Predictor is illustrated in Figure 7, handling 3D data (time, height, and width) to accommodate the dynamic nature of videos.

This U-VIT architecture specialized for Video Visible Difference Predictor can adeptly handle 3D data consisting of time, height, and width to accommodate the intricacies of video dynamics. Beginning with two distinct input streams, the reference and distorted video volumes are initially processed by the spatio-temporal display model, laying the foundation for their progression through the network. Both streams encounter a patch embedding layer early on, ensuring the input is a series of spatio-temporal tokens. As the streams delve deeper into the encoding branch of the U-ViT, they undergo a systematic sequence of transformer encoder stage. Each of transformer encoder stage is complemented by a duo of two transformer encoder blocks, where the first block performs spatial attention and the second performs temporal attention. This design choice facilitates the extraction of multi-scale features while adeptly learning the spatial and temporal features.

At the architecture's midpoint, the processed streams from both branches converge. In the middle stage, the combined features of both streams are further refined with one transformer encoder stage, ensuring a cohesive merge. This merged representation then embarks on the decoding branch of the U-ViT. Similar with the encoding branch, the decoding branch consists of a series of transformer encoder stage. At each stage, spatial attention and temporal attention work in tandem to refine and reconstruct the video volume. The culmination of this intricate process is the final few layers where the data, processed through an additional linear layer, is mapped to its final output format via a 1x1 Conv3D layer. This output delineates the visible differences in the videos, spotlighting perceptual discrepancies between the reference and distorted inputs. By synergizing the U-ViT's U-shaped structure with spatio-temporal attention mechanisms, this architecture achieves remarkable proficiency in capturing and representing subtle differences in video content.

Our architecture excels in video Visible Difference Predictor (VDP) due to its intrinsic ability to comprehend both spatial and temporal context within video data. Its U-shaped architecture empowers it to capture and refine features across varying levels of detail, making it adept at identifying visible differences in video content. This is further enhanced by its multi-scale feature extraction capability through spatio-temporal attention mechanisms. The incorporation of attention allows the model to focus on key regions, crucial for pinpointing areas where differences manifest. U-ViT's resilience, facilitated by residual connections, enables the handling of deep networks, necessary for capturing intricate patterns within video data. Consequently, U-ViT architectures deliver state-of-the-art results in VDP, which can be demonstrated in Section V.

V. Experiment

A. Experimental Setup

For training Deep Video Visible Difference Metric (DV2DM), we employ the probability loss function proposed by [44]. The loss function offers a crucial method to model the experiment data, where it conceptualizes the human annotation process as a sequence of discovery, attention and detection activities to capture the uncertainty in the human-annotated dataset.

Following the pre-processing stage, we split the video into numerous non-overlapping patches and remove the patches where there is no difference between reference patch and distortion patch. Each patch consists of 12 frames and the size of each frame is 48x48.

We implement DV2DM on the PyTorch v2.0 framework [26] and utilized MMEngine v0.9.0 [8] to perform distributed training on 8 NVIDIA Ampere A100 GPUs. During the training process, we use the adaptive moment estimation with weight decay (AdamW) [15] optimizer with a learning rate of 1×10^{-4}. We partition the training process into two stages, as explained below.

TABLE II
CROSS-VALIDATION RESULT.

	All	P30L110	P30L200	P30L300	P50L110	P50L200	P50L300	P60L110	P60L200	P60L300
MSE	12.8 ± 3.09	13.3 ± 2.89	13.4 ± 2.68	14.3 ± 3.12	13.8 ± 3.35	14.5 ± 3.34	13.2 ± 3.31	14.1 ± 2.92	13.6 ± 3.34	13.7 ± 3.22
PSNR	6.19 ± 4.48	6.04 ± 4.48	6.08 ± 4.48	5.18 ± 4.54	5.32 ± 4.45	5.15 ± 4.57	5.20 ± 4.53	4.98 ± 4.58	5.15 ± 4.55	5.44 ± 4.52
SSIM	4.45 ± 1.40	4.89 ± 1.06	4.83 ± 1.26	3.94 ± 1.26	4.10 ± 1.57	4.21 ± 1.07	3.72 ± 1.23	3.92 ± 1.37	3.80 ± 1.15	4.18 ± 1.40
PU MSE	12.9 ± 3.06	13.4 ± 2.80	13.4 ± 2.59	14.5 ± 3.14	14.0 ± 3.44	14.7 ± 3.41	13.4 ± 3.14	14.2 ± 2.83	13.8 ± 3.19	13.9 ± 3.21
PU_PSNR	5.91 ± 4.40	6.00 ± 4.24	5.94 ± 4.32	4.92 ± 4.48	5.11 ± 4.41	5.09 ± 4.40	4.85 ± 4.49	4.77 ± 4.53	4.93 ± 4.46	5.24 ± 4.43
PU_SSIM	5.02 ± 1.25	4.97 ± 1.29	5.12 ± 1.38	4.19 ± 1.31	4.50 ± 1.44	4.15 ± 1.20	4.00 ± 1.15	3.85 ± 1.28	3.81 ± 1.15	4.43 ± 1.39
DPVM	13.4 ± 4.13	13.3 ± 3.91	13.7 ± 4.19	13.1 ± 4.08	13.3 ± 4.39	13.6 ± 4.09	12.0 ± 4.00	12.4 ± 4.17	12.6 ± 3.96	13.1 ± 4.21
FovVideoVDP	4.06 ± 1.05	3.99 ± 0.73	4.28 ± 1.05	3.30 ± 0.73	3.55 ± 0.73	3.49 ± 0.71	3.72 ± 1.18	3.80 ± 1.18	3.85 ± 1.07	3.72 ± 0.94
DV2DM	$\mathbf{2.59 \pm 1.23}$	$\mathbf{2.59 \pm 1.21}$	$\mathbf{2.43 \pm 0.65}$	$\mathbf{2.43 \pm 0.83}$	$\mathbf{2.60 \pm 0.77}$	$\mathbf{2.69 \pm 1.07}$	$\mathbf{2.73 \pm 0.74}$	$\mathbf{2.72 \pm 0.80}$	$\mathbf{2.96 \pm 0.95}$	$\mathbf{2.63 \pm 0.91}$

B. Training Details

1) Pre-training Stage: Due to the limited variations in luminance, viewing distance and FPS in the manually annotated dataset, we initially conduct our training process on the synthetic dataset containing over 1 million patches. The generation of the synthetic dataset is described in Section III-C. Although the labels generated by the white-box visibilty metric FovVideoVDP [19] may not be entirely accurate, they characterize the general relationship between the original data and labels under different observation environments, aiding the model in capturing these relationships more effectively, which may be lacking in the manually annotated dataset.

2) Fine-tuning Stage: In this stage, we initialize the model's weights with pre-trained weights and fine-tune it using manually annotated data. The data processing workflow during fine-tuning stage remains consistent with the pre-training stage.

C. Result

To evaluate the performance of our model and minimize the bias introduced by training-test split, we split the annotated data into 5 folds and ensure that each data point belongs to only one fold to perform 5-fold cross-validation. We report the mean and standard deviation of the likelihood used for the loss function after applying the negative log transformation. A higher likelihood corresponds to a smaller negative log-likelihood, indicating a higher accuracy.

We compare DV2DM with other methods on the manually annotated data. The compared methods range from traditional metrics such as MSE, PSNR and SSIM to metrics employing perceptually uniform encoding such as PU-MSE, PU-PSNR and PU-SSIM [46]. In addition, our comparsion includes the deep image VDP, DPVM and the unique but training-free video VDP, FovVideoVDP. Since the manually annotated data includes three difference viewing distances and three different screen luminances, the result of cross-validataion is shown separately in Table II.

According to the result, similar trends are observed among different methods when evaluating their cross-validation results over the entire test set and its subsets. Notably, compared to other approaches, DV2DM exhibits the lowest mean and standard deviation of negative log-likelihood values across the whole dataset and all subsets. This outcome indicates that DV2DM consistently performs well under various viewing conditions. Additionally, it should be noted that some fluctuations occur in the performance of all methods across different subsets, which is mainly attributed to variations in video content between these subsets. While SSIM and PU-SSIM seemed to yield better outcomes according to the table, they lacked responsiveness towards factors like PPD and video frame rate. Consequently, ablation experiments were conducted to clarify this phenomenon.

D. Ablation Study

In order to examine whether DV2DM can account for the change of view conditions such as viewing distance, luminance and frames per second, we conduct a series of ablation experiments and report the results in Figure 8. We vary the view conditions for the manual annotation process on identical video contents.

Based on the results shown in Figure 8, it is obvious that MSE provides little prediction for distortions in the videos, consequently leading to higher negative log-likelihood levels. Both PSNR and PU-PSNR methods map pixel differences into the logarithmic domain, resulting in perceived difference visibility heatmaps with inferior visual effects compared to other approaches.

For SSIM and PU-SSIM, which perform better according to Table II, although they exhibit some similarity to human annotations, their predictions do not align with the perception process of the human visual system. As illustrated in the PPD comparison (Figure 8, rows 5 and 6), both SSIM and PU-SSIM assign high predicted values to background walls; however, this region is high-frequency areas where human eyes struggle to perceive differences. The absence of background annotations in human-labeled data further supports the argument that SSIM and PU-SSIM predictions deviate from human visual perception processes.

For the DPVM method, while accounting for PPD and luminance impacts on predictive outcomes, its application as an image VDP fails to consider frame rate specific to videos. Additionally, DPVM's conservative predictions lead to overall elevated negative log-likelihood levels.

Both FovVideoVDP and DV2DM provide corresponding feedback for different viewing distances, luminance, and FPS, where the general trends are coordinated, while DV2DM yields explicitly better predictions. Furthermore, despite experiencing overestimation issues, since these two methods generally reflect human perception outcomes accurately, practical solutions like raising threshold values may alleviate concerns related to overestimation.

More ablation results are shown as Figure 13, Figure 14 and Figure 15.

Fig. 8. Reference images, labels and metrics' predictions examples from the manually annotated data. NLL represents negative-log-likelihood.

VI. Applications

A. Content-adaptive Watermarking

With the advance in the Artificial Intelligence Generated Content (AIGC) technology, numerous images and videos are used to train AIGC algorithms without authorization from the authors. To protect the copyright of the authors, embedding watermarks stands as a pivotal method. In the case of images or videos, a watermark is often a logo or text that is almost imperceptible but nevertheless real. Hence, the foundation of our application lies in utilizing DV2DM to detect the correct regions for watermark embedding and make the watermark undetectable to the human eye.

In order to embed a watermark, we add a grid of watermark patches in 64x64 pixels to the first frame of the original video. Determining the watermark's intensity involves employing a grid search method. We start at a high watermark intensity so that the difference is clearly visible. Then we iteratively reduce the intensity until DV2DM indicates that the maximum value of all patches for the first frame falls below a predefined threshold. This process is repeated across all the frames to generate the watermarked video.

Although DV2DM has not been trained specifically for this distortion, the watermark added under the guidance of DV2DM remains imperceptible, as shown in the Figure 9. The results not only demonstrate DV2DM can detect the suitable regions for watermark addition and guide the watermarking process, but also highlight its adaptability to distortion types absent in the training set. Furthermore, we provide the results of watermarks multiplied by a constant to offer a clearer view of the variations in watermark intensities. For example, the clothing region, where there is a more noticeable color variation and higher frequency, leading to a higher watermark intensity. The intensity remains small in the low-frequency regions.

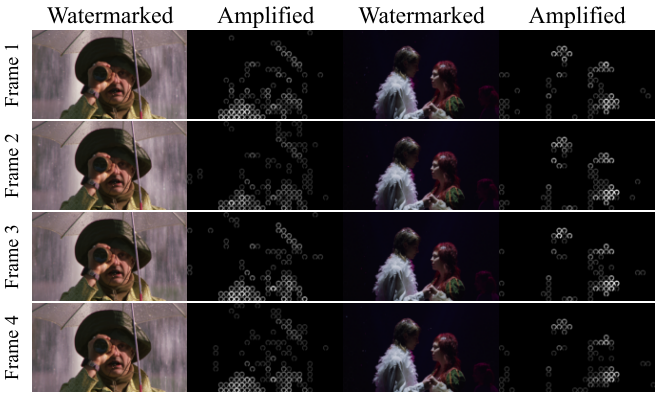

Fig. 9. Examples of content-adaptive watermark application.

B. Visually Lossless Video Compression

During the video compression process, inappropriate compression parameters can lead to excessively large file sizes or poor video quality. To balance the trade-off between file size and video quality, visually lossless video compression minimizes file sizes by making the distortion caused by compression practically imperceptible to the human eye. Achieving visually lossless video compression can depend on using appropriate compression parameters, although determining whether a video is visually lossless can pose a challenge.

Hence, we employ HEVC as our video codec, Constant Rate Factor (CRF) as the compression parameter to optimize. and

use DV2DM to determine if the compressed video is visually lossless. Beginning with the highest CRF, which renders the video quality unacceptable. We feed the compressed video into DV2DM to generate the distortion heatmap. By averaging the heatmap for each frame, if it is larger than the predefined threshold, we decrease the CRF, improving video quality while increasing file size. This iterative process continues until the averaged heatmap value falls below the threshold. The frames pairs are shown as Figure 10.

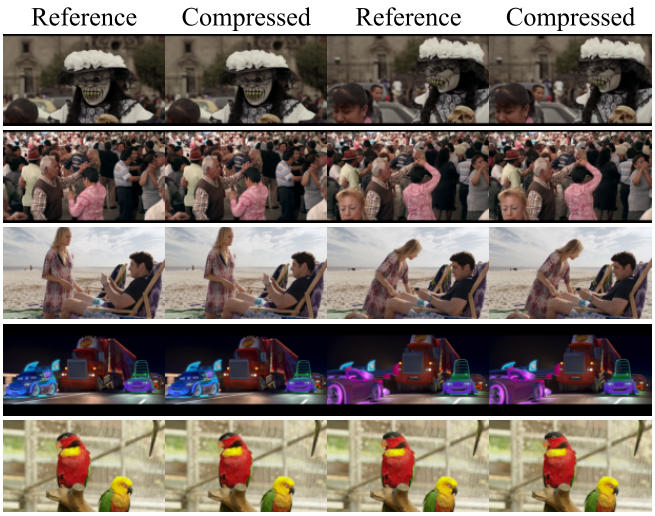

Fig. 10. The frames pairs sampled from the original videos and visually lossless compressed videos.

We set the threshold to 0.01, namely 99% of observers are unable to detect the differences between the compressed videos and the original videos. After obtaining the compressed videos, we recruit 15 volunteers to conduct a Two-Alternative Forced-Choice (2AFC) experiment, where volunteers are asked if they could discern any differences between the compressed and original videos. Based on the 2AFC experiment results, 95% of participants cannot perceive any distortion between the two videos. Although 95% is less than 99%, our DV2DM demonstrates some overestimation. Nevertheless, the results highlight the potential of DV2DM guiding the visually lossless video compression.

C. Invisible Adversarial Attacks

In the adversarial attack task, it is necessary to apply imperceptible noise to the input data to induce errors in the predictions of the model. However, a too low noise intensity might fail to allow the model to produce erroneous results while an intense noise may be detected by the human. Therefore, we need to maximize the noise intensity while it is almost imperceptible to the human eye, thereby achieving the invisible adversarial attack.

Thus, we utilize our DV2DM to detect whether humans can perceive the noise in the task of adversarial attack on the video models. We use Multi-scale Vision Transformers (MViT) [51], [52] as the target video classification model and employ Fast Gradient Sign Method (FGSM) [50] as the adversarial attack approach. We start with a intense noise and iteratively reduce the noise intensity until the maximum value of the heatmap predicted by DV2DM is less than the threshold. The frames sampled from the input video and the corresponding predictions of the model before and after the adversarial attack and are shown as Figure 11.

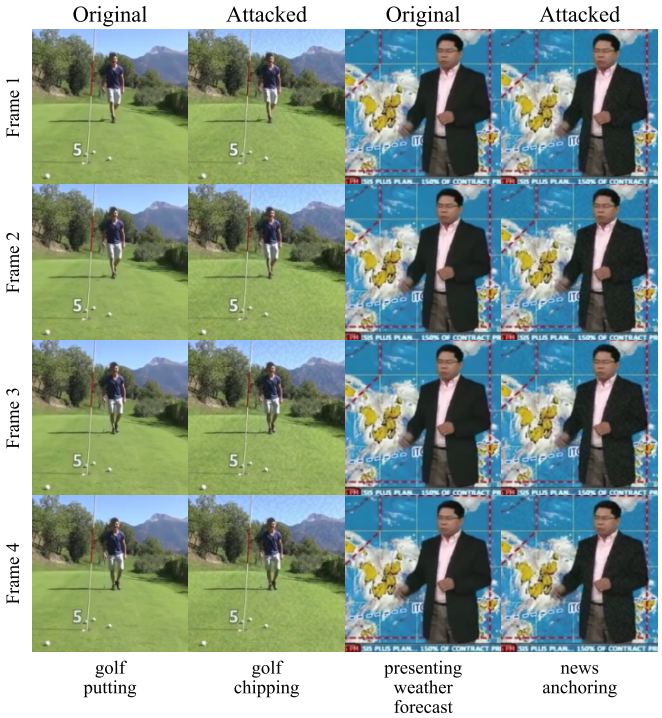

Fig. 11. Frame samples and predictions of the model before and after the adversarial attack.

From the Figure 11, it is clear that there are errors in the predictions of the model, while we cannot perceive the noise in the image after being adversarial attacked. The results indicat that utilizing DV2DM to predict whether the noise is perceivable in the process of undetectable adversarial attack is effective.

D. Video Super-Resolution Metric

In the field of video research, the video super-resolution is also an essential task. It aims to convert low-resolution video to high-resolution while minimizing the distortion caused by the super-resolution algoithms. To evaluate the video quality after super-resolution, Mean Squared Error (MSE) and Structural Similarity (SSIM) are often used as the metrics. However, both MSE and SSIM are which are evaluated at the pixel level which is different from the perception process of the human eye. Thus MSE and SSIM are difficult to exactly and accurately describe perception quality. Therefore, we adopt DV2DM as an evaluation metric for the video super-resolution task to assess the distortion introduced by the super-resolution process.

We downsample the video with a resolution of 3840x2160 by ffmpeg before restoring it to the original resolution by an interpolation algorithm. Then we predicte the distortion

between these two videos using MSE, SSIM and DV2DM respectively and get the predicted heatmap. The results are shown as Figure 12.

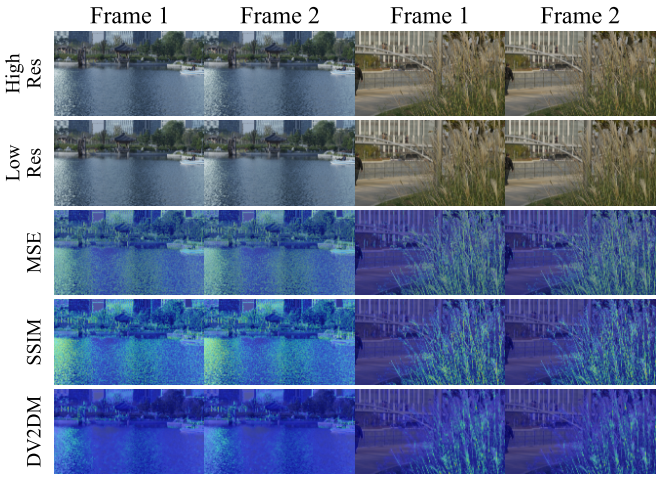

Fig. 12. The heatmaps predicted by Mean Squared Error (MSE), Structural Similarity (SSIM) and DV2DM based on high-resolution frames and low-resolution frames after super-resolution.

From the Figure 12, it can be seen that MSE, SSIM and DV2DM keep the consistency in the prediction trend, which shows the reasonableness of DV2DM as an evaluation metric for the video super-resolution task. Meanwhile, it can be noticed that the heatmap predicted by DV2DM is smoother compared with MSE and SSIM, which is more in line with the human eye's perception of video quality. Therefore, DV2DM can be used as an evaluation metric for the video super-resolution task to simulate the human eye's perception of the video after super-resolution.

VII. Conclusion

VDP within the realm of video processing remains an underexplored domain, presenting multiple challenges, including a shortage of datasets and the distinct temporal and spatial attributes of video data. This research endeavors to investigate the applicability of VDP to video content, along with examining the influence of various viewing conditions such as viewing distance, luminance and frame rate, which are crucial considerations. We introduce a two-stage model, the Deep Video Visible Difference Metric (DV2DM), establish a comprehensive dataset named ViLocVis and provide several applications based on the DV2DM. Our empirical findings demonstrate that the ViLocVis dataset accurately embodies the anticipated VDP experience, while the DV2DM proficiently captures the essence of video VDP, reliably forecasting perceptible disparities across diverse viewing scenarios. Ultimately, this study addresses the void in video VDP research, offering both a dataset and a framework that serve as valuable references for future explorations in this burgeoning field.

References

[1] Ahn, S., Choi, Y., and Yoon, K. Deep learning-based distortion sensitivity prediction for full-reference image quality assessment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 344–353, 2021.

[2] Akyazi, P. and Ebrahimi, T. Learning-based image compression using convolutional autoencoder and wavelet decomposition. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, number CONF, 2019.

[3] Aydin, T. O., Mantiuk, R., Myszkowski, K., and Seidel, H.-P. Dynamic range independent image quality assessment. *ACM Trans. Graph.*, 27(3): 1–10, aug 2008. ISSN 0730-0301. doi: 10.1145/1360612.1360668. URL https://doi.org/10.1145/1360612.1360668.

[4] Banterle, F., Artusi, A., Moreo, A., and Carrara, F. Nor-vdpnet++: Efficient training and architecture for deep no-reference image quality metrics. In *ACM SIGGRAPH 2021 Talks*, pp. 1–2. 2021.

[5] Charrier, C., Lebrun, G., and Lezoray, O. A machine learning-based color image quality metric. In *CGIV*, pp. 251–256, 2006.

[6] Chen, C., Shi, Y. Q., and Su, W. A machine learning based scheme for double jpeg compression detection. In *2008 19th international conference on pattern recognition*, pp. 1–4. IEEE, 2008.

[7] Cheon, M. and Lee, J.-S. Subjective and objective quality assessment of compressed 4k uhd videos for immersive experience. *IEEE Transactions on Circuits and Systems for Video Technology*, 28(7):1467–1480, 2018. doi: 10.1109/TCSVT.2017.2683504.

[8] Contributors, M. MMEngine: Openmmlab foundational library for training deep learning models. 2022.

[9] Daly, S. J. Visible differences predictor: an algorithm for the assessment of image fidelity. In *Human Vision, Visual Processing, and Digital Display III*, volume 1666, pp. 2–15. SPIE, 1992.

[10] Illumination, I. Recommendations on uniform color spaces. *Color-Difference Equations, Psychometric Color Terms, CIE*, 1978.

[11] Ji, R., Karam, L. J., et al. Learning-based visual compression. *Foundations and Trends® in Computer Graphics and Vision*, 15(1):1–112, 2023.

[12] Kim, W., Kim, J., Ahn, S., Kim, J., and Lee, S. Deep video quality assessor: From spatio-temporal visual sensitivity to a convolutional neural aggregation network. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 219–234, 2018.

[13] Li, T., Min, X., Zhao, H., Zhai, G., Xu, Y., and Zhang, W. Subjective and objective quality assessment of compressed screen content videos. *IEEE Transactions on Broadcasting*, 67(2):438–449, 2020.

[14] Liu, W., Duanmu, Z., and Wang, Z. End-to-end blind quality assessment of compressed videos using deep neural networks. In *ACM Multimedia*, pp. 546–554, 2018.

[15] Loshchilov, I. and Hutter, F. Fixing weight decay regularization in adam. *CoRR*, abs/1711.05101, 2017. URL http://arxiv.org/abs/1711.05101.

[16] Lu, G., Zhong, T., Geng, J., Hu, Q., and Xu, D. Learning based multi-modality image and video compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6083–6092, 2022.

[17] Mantiuk, R., Daly, S. J., Myszkowski, K., and Seidel, H.-P. Predicting visible differences in high dynamic range images: model and its calibration. In *Human Vision and Electronic Imaging X*, volume 5666, pp. 204–214. SPIE, 2005.

[18] Mantiuk, R., Kim, K. J., Rempel, A. G., and Heidrich, W. Hdr-vdp-2: A calibrated visual metric for visibility and quality predictions in all luminance conditions. *ACM Transactions on graphics (TOG)*, 30(4):1–14, 2011.

[19] Mantiuk, R. K., Denes, G., Chapiro, A., Kaplanyan, A., Rufo, G., Bachy, R., Lian, T., and Patney, A. Fovvideovdp: A visible difference predictor for wide field-of-view video. *ACM Transactions on Graphics (TOG)*, 40 (4):1–19, 2021.

[20] Mercat, A., Viitanen, M., and Vanne, J. Uvg dataset: 50/120fps 4k sequences for video codec analysis and development. In *Proceedings of the 11th ACM Multimedia Systems Conference*, MMSys '20, pp. 297–302, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450368452. doi: 10.1145/3339825.3394937. URL https://doi.org/10.1145/3339825.3394937.

[21] Min, X., Zhai, G., Zhou, J., Farias, M. C. Q., and Bovik, A. C. Study of subjective and objective quality assessment of audio-visual signals. *IEEE Transactions on Image Processing*, 29:6054–6068, 2020. doi: 10.1109/TIP.2020.2988148.

[22] Nadenau, M. J. and Reichel, J. Compression of color images with wavelets considering the hvs. In *Human Vision and Electronic Imaging IV*, volume 3644, pp. 129–140. SPIE, 1999.

[23] Narwaria, M. and Lin, W. Svd-based quality metric for image and video using machine learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 42(2):347–364, 2011.

[24] Narwaria, M., Mantiuk, R. K., Da Silva, M. P., and Le Callet, P. Hdr-vdp-2.2: a calibrated method for objective quality prediction of high-dynamic range and standard images. *Journal of Electronic Imaging*, 24 (1):010501–010501, 2015.

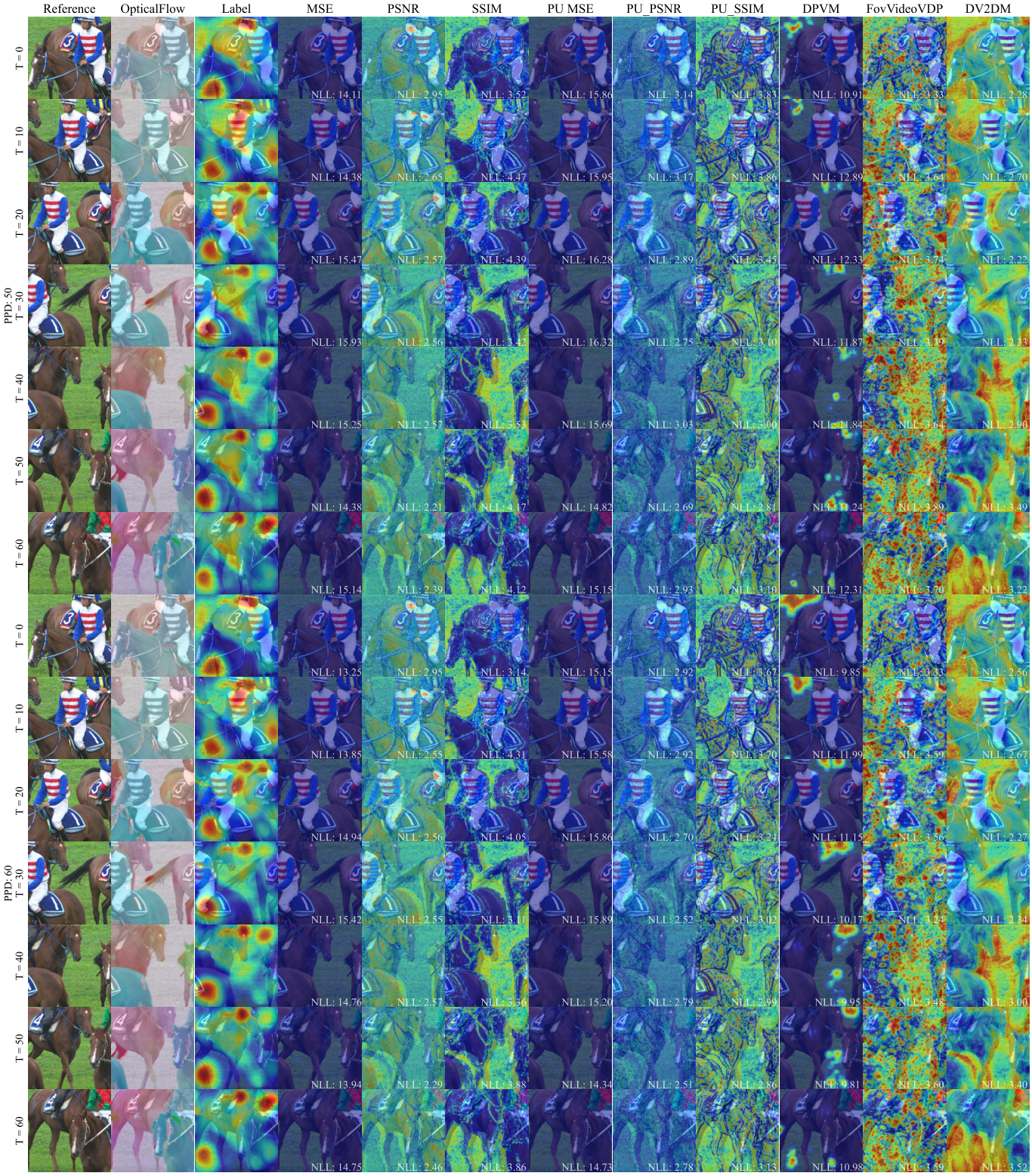

Fig. 13. Reference images, labels and metrics' predictions examples under different PPDs.

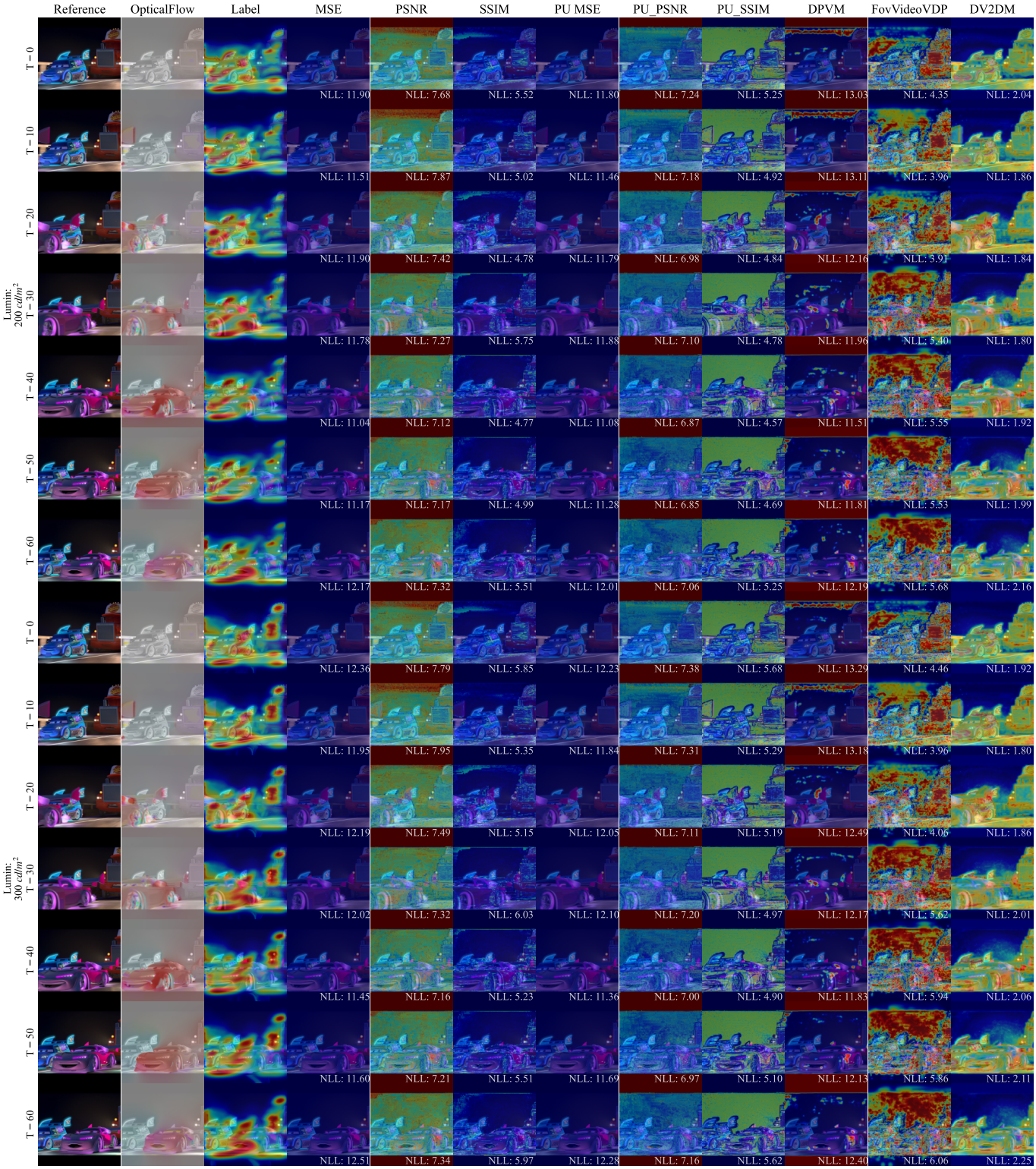

Fig. 14. Reference images, labels and metrics' predictions examples under different luminances.

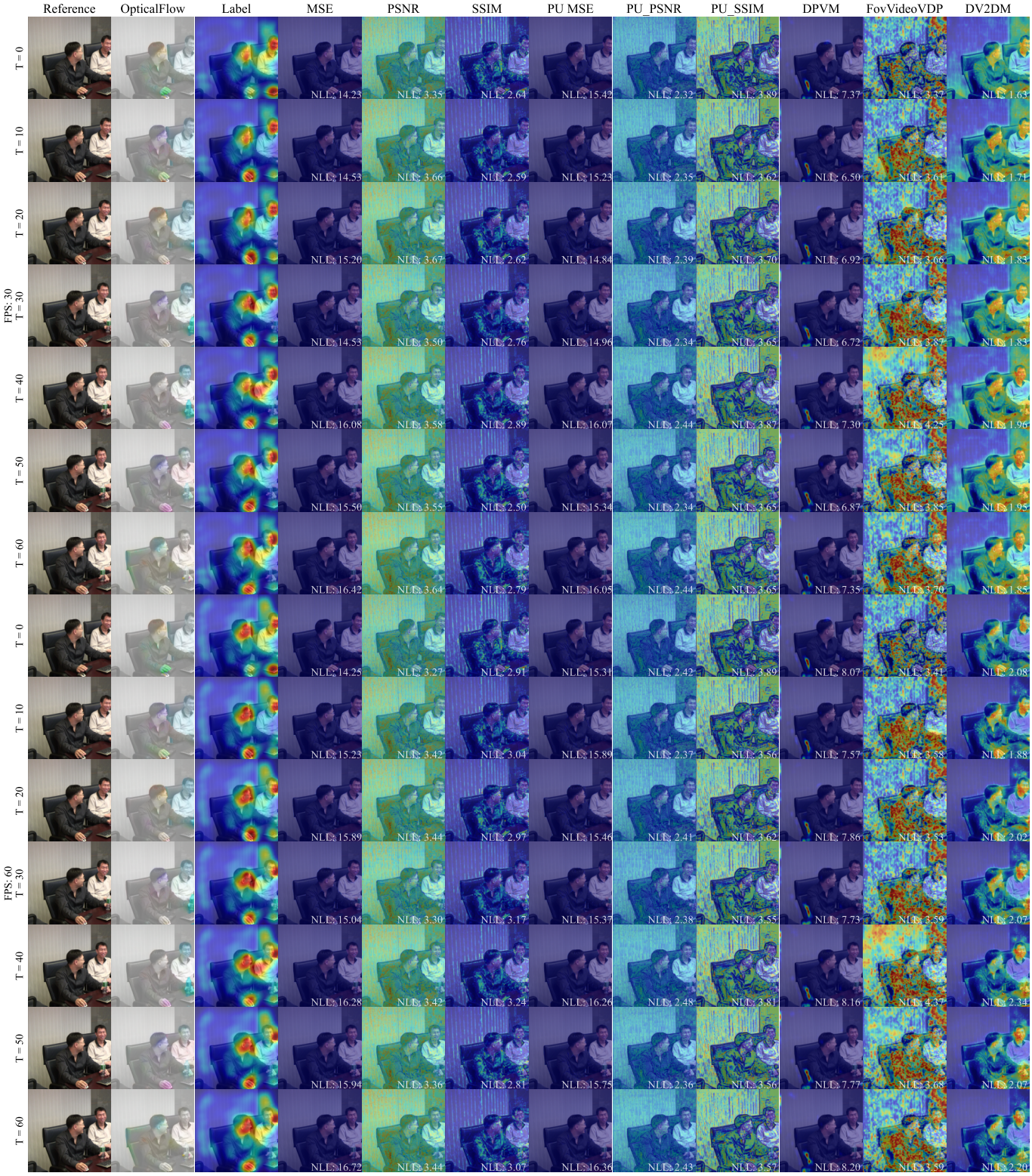

Fig. 15. Reference images, labels and metrics' predictions examples with different FPSs.

[25] Palmer, S. E. *Vision science: Photons to phenomenology*. MIT press, 1999.

[26] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché Buc, F., Fox, E., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 32*, pp. 8024–8035. Curran Associates, Inc., 2019.

[27] Peli, E. *Vision Models for Target Detection and Recognition: In Memory of Arthur Menendez*, volume 2. World scientific, 1995.

[28] Pöppel, E. A hierarchical model of temporal perception. *Trends in cognitive sciences*, 1(2):56–61, 1997.

[29] Selfridge, R., Noland, K. C., and Hansard, M. Visibility of motion blur and strobing artefacts in video at 100 frames per second. In *Proceedings of the 13th European Conference on Visual Media Production (CVMP 2016)*, pp. 1–10, 2016.

[30] Shahid, M., Rossholm, A., and Lövström, B. A no-reference machine learning based video quality predictor. In *2013 Fifth International Workshop on Quality of Multimedia Experience (QoMEX)*, pp. 176–181. IEEE, 2013.

[31] Shahid, M., Rossholm, A., Lövström, B., and Zepernick, H.-J. No-reference image and video quality assessment: a classification and review of recent approaches. *EURASIP Journal on image and Video Processing*, 2014:1–32, 2014.

[32] Shang, Z., Ebenezer, J. P., Bovik, A. C., Wu, Y., Wei, H., and Sethuraman, S. Assessment of subjective and objective quality of live streaming sports videos. In *2021 Picture Coding Symposium (PCS)*, pp. 1–5, 2021. doi: 10.1109/PCS50896.2021.9477502.

[33] Shang, Z., Ebenezer, J. P., Wu, Y., Wei, H., Sethuraman, S., and Bovik, A. C. Study of the subjective and objective quality of high motion live streaming videos. *IEEE Transactions on Image Processing*, 31:1027–1041, 2022. doi: 10.1109/TIP.2021.3136723.

[34] Song, L., Tang, X., Zhang, W., Yang, X., and Xia, P. The sjtu 4k video sequence dataset. In *2013 Fifth International Workshop on Quality of Multimedia Experience (QoMEX)*, pp. 34–35, 2013. doi: 10.1109/QoMEX.2013.6603201.

[35] Sreelekha, G. and Sathidevi, P. An hvs based adaptive quantization scheme for the compression of color images. *Digital Signal Processing*, 20(4):1129–1149, 2010.

[36] Sullivan, G. J., Ohm, J.-R., Han, W.-J., and Wiegand, T. Overview of the high efficiency video coding (hevc) standard. *IEEE Transactions on Circuits and Systems for Video Technology*, 22(12):1649–1668, 2012. doi: 10.1109/TCSVT.2012.2221191.

[37] Wang, H., Gan, W., Hu, S., Lin, J. Y., Jin, L., Song, L., Wang, P., Katsavounidis, I., Aaron, A., and Kuo, C.-C. J. Mcl-jcv: A jnd-based h.264/avc video quality assessment dataset. In *2016 IEEE International Conference on Image Processing (ICIP)*, pp. 1509–1513, 2016. doi: 10.1109/ICIP.2016.7532610.

[38] Wang, Z., Simoncelli, E. P., and Bovik, A. C. Multiscale structural similarity for image quality assessment. In *The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*, volume 2, pp. 1398–1402. Ieee, 2003.

[39] Wang, Z., Bovik, A. C., Sheikh, H. R., and Simoncelli, E. P. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

[40] Watson, A. B. High frame rates and human vision: A view through the window of visibility. *SMPTE Motion Imaging Journal*, 122(2):18–32, 2013.

[41] Watson, A. B., Hu, J., and McGowan, J. F. Digital video quality metric based on human vision. *Journal of Electronic imaging*, 10(1):20–29, 2001.

[42] Winkler, S. Perceptual video quality metrics—a review. *Digital video image quality and perceptual coding*, pp. 155–180, 2017.

[43] Winkler, S. and Mohandas, P. The evolution of video quality measurement: From psnr to hybrid metrics. *IEEE transactions on Broadcasting*, 54(3):660–668, 2008.

[44] Wolski, K., Giunchi, D., Ye, N., Didyk, P., Myszkowski, K., Mantiuk, R., Seidel, H.-P., Steed, A., and Mantiuk, R. K. Dataset and metrics for predicting local visible differences. *ACM Transactions on Graphics (TOG)*, 37(5):1–14, 2018.

[45] Ye, N., Wolski, K., and Mantiuk, R. K. Predicting visible image differences under varying display brightness and viewing distance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5434–5442, 2019.

[46] R. K. Mantiuk and M. Azimi. PU21: A novel perceptually uniform encoding for adapting existing quality metrics for HDR, *2021 Picture Coding Symposium (PCS)*

[47] Zhang, L., Zhang, L., Mou, X., and Zhang, D. Fsim: A feature similarity index for image quality assessment. *IEEE transactions on Image Processing*, 20(8):2378–2386, 2011.

[48] Zhang, R., Isola, P., Efros, A. A., Shechtman, E., and Wang, O. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 586–595, 2018.

[49] Zhang, X., Wandell, B. A., et al. A spatial extension of cielab for digital color image reproduction. In *SID international symposium digest of technical papers*, volume 27, pp. 731–734. Citeseer, 1996.

[50] Goodfellow, I. J., Shlens, J., and Szegedy, C. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

[51] Fan, H., Xiong, B., Mangalam, K., Li, Y., Yan, Z., Malik, J., and Feichtenhofer, C. Multiscale vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 6824–6835, 2021.

[52] Li, Y., Wu, C.-Y., Fan, H., Mangalam, K., Xiong, B., Malik, J., and Feichtenhofer, C. Mvitv2: Improved multiscale vision transformers for classification and detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4804–4814, 2022.

[53] Rosenblatt, M. Remarks on Some Nonparametric Estimates of a Density Function. *The Annals of Mathematical Statistics*, 27(3):832 – 837, 1956.

[54] Parzen, E. On Estimation of a Probability Density Function and Mode. *The Annals of Mathematical Statistics*, 33(3):1065 – 1076, 1962.