Virtual See-Through

Jean Mabillard, *Student*, Damien Perritaz and Christophe Salzmann, *Assistants*, Denis Gillet, *MER*
Master Project, *Automatic Control Laboratory*, EPFL
23 February 2007

Abstract—Network transmissions are subjected to bandwidth constraint. In this context, a method is proposed to adapt the video encoding parameters to follow bit rate constraint fluctuations while maximizing the user *Quality of Perception* (QoP). Once the video codec system is identified for selected encoding parameters, a perception model is built using subjective tests methodology, where subjects are asked to evaluate their perception of the video stream for several encoding parameters. The perception model corresponds to an *Optimal Adaptation Path* (OAP) in the parameters space. The model proposed is developed for the chemical plant context. A closed-loop control system which takes advantage of the perception model is proposed to control the codec bit rate by modifying frame rate and quality parameters. The controller aims at compensating modeling error and at rejecting video perturbations generated by the video content itself.

Index Terms—Video Quality, User Perception, Codec, Quality Evaluation, Adaptation, Control, Perception Model

1 Introduction

This project is a part of a research project called 6^{th} *Sense* whose goal is to develop a wearable supervision system for industrial plants. This global project starts with this observation: currently, in industrial chemical context, manipulations on a plant require the collaboration of two operators, one located in a control room and the other one near the installation, each one communicating with a radio. In order to offer more liberty to the operator, the aim of 6^{th} *Sense* is to develop a hands-free interface system, which will consist of see-through glasses augmented with virtual data. By extension, the system must be able to transmit audio and video streams between several colleagues through a wireless network to permit a collaborative work. Nevertheless best-effort networks are unreliable and unpredictable. Many factors can affect the quality of a transmission, usually denoted as *Quality of Service* (QoS), such as delay or loss. Adaptation techniques are used to reduce network congestion and packet loss by matching the rate of the data stream to the available network bandwidth. Without adaptation, any data transmitted exceeding the available bandwidth could be buffered, discarded or lost in the network. This has a negative impact on the quality of the received stream.

The *Virtual See-Through* project focuses on video transmission and particularly on adapting the video encoder parameters to respect the bandwidth constraint while maximizing the user *Quality of Perception* (QoP). Video quality adaptation mechanisms generally indicate how the bit rate of a video should be adjusted in response to a network variation, but does not address the video perception. In the same way video quality evaluation measures the quality of the video as perceived by the users, but is not designed for adaptive transmissions. The goal of this project is to link these two concepts.

Figure 1 presents the context of this work. *Virtual See-Through* is implemented with a *Head Mounted Display* (HMD) and a camera. The video stream is displayed on the glasses to emulate transparency and transmission is simulated with the passage of data through an encoder-decoder (codec). It can be used to zoom or superpose information in *Augmented Reality* applications.

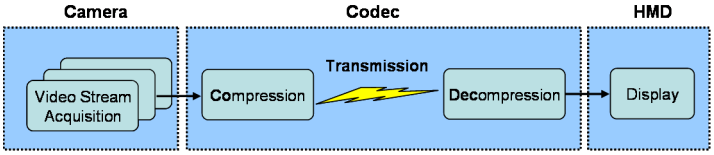

Fig. 1. Project Context Diagram

In order to follow the bit rate, which can be seen as a reference trajectory, two adjustable encoding parameters are chosen: the *frame rate* and the *compression rate*. Hence, the system not only has to satisfy the bandwidth in real time but it also should give the best possible perception quality for the chosen parameters. This impose to find a perception model based on the *Human Visual System* (HVS), which shall provide the optimal encoding parameters for a given bit rate. In term of closed-loop control the system can be seen as presented in Figure 2.

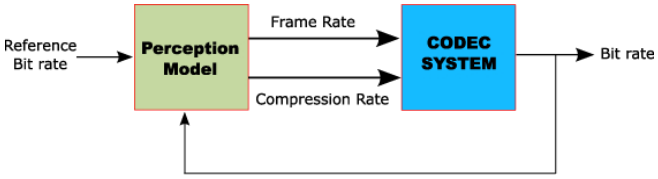

Fig. 2. Closed-loop control system

From this point of view, the project can be divided into three axes:

1) Codec system
2) Perception model
3) Control

The rest of the paper is organized as follows: Section 2 introduces the general concept of video compression, gives an overview of video acquisition and proposes an identification of the codec system. Section 3 explains how to achieve video

quality evaluation to find a perception model based on *Human Visual System*. Section 4 exposes the concept of closed-loop control system based on a perception model. Conclusions and directions for future work are presented in Section 5.

2 CODEC SYSTEM

Video acquisition is performed on a macintosh with an *iSight* webcam. The *iSight* camera generates 640x480 YUV422[1] video stream at 30 frames per second (fps) giving a row transmission rate of 150Mbps[2]. Figure 3 compares bit rates of several standard transmission protocols [3] [4]. It can be deduce from this table that raw video is to large for wireless applications[3], besides the fact that network sharing decreases the information bit rate. Thus video compression is required to allow data transmission.

Interface		Bitrate (Mbps)
Firewire 400		400
Firewire 800		800
USB, Low speed		1.5
USB, Full speed		12
USB, High speed		480
WLAN	Wi-fi (IEEE 802.11b)	11
	IEEE 802.11g	54

Fig. 3. Standard transmission protocols bit rates

2.1 Standards of Video Compression and Decompression

There are two main standards for encoding video content. These are the ITU (International Telecommunication Union) [5] and the MPEG (Motion Picture Experts Group) [6]. A complete list of codecs can be found in [7].

MPEG standards

- MPEG-1(1993): Defined for relatively low bit rate coding (1.5Mbps) of low spatial resolution pictures. It is a popular standard of compression for VideoCD (VCD) and video on the Internet. MPEG-1 consists of several parts and layers and part 3 layer 3 corresponds to the most popular standard for digital compression of audio known as MP3 and often confused with MPEG-3.
- MPEG-2(1995): Also known as H.262, it follows the collaborative work of MPEG and ITU to address a wide variety of applications such as Digital Broadcast Television (DBT), high definition television (HDTV) and DVD compression. It is designed for bit rates between 1.5 and 15Mbps.
- MPEG-4(1998/2002): Designed for very low to very high bit rates, it can be used for internet and wireless applications. MPEG-4 innovates with object-based compression, which allows individual objects within a scene to be tracked separately and compress together resulting in very efficient compression. Well-known codecs such as Divx or Xvid are different implementations of MPEG-4 Part 2.

ITU standards

- H.261(1990): It is the first practical digital video coding standard. All subsequent international video coding standards have been based closely on its design. It is used primarily in older videoconferencing products and designed for transmission over ISDN lines on which data rates are multiples of 64kbps (up to 2Mbps).
- H.263(1996): Targeted at videoconferencing applications, this codec provides a suitable replacement for H.261 at all bit rates.
- H.264(2002): Also known as MPEG-4 part 10 or AVC, this emerging new standard extends MPEG-4 coding by increasing the compression ratio and increasing video quality. It can be applied to a very wide variety of applications (for both low and high bit rates and low and high resolution). H.264 has recently been adopted into a number of company products, for example Playstation, iPod or HD DVD/Blu-ray Disc.

2.2 Principle of Video Coding

This section presents an overview of video coding mechanism explained in [8][9][10]. Video coding is based on the fact that there is a strong correlation between both successive picture frames and within the picture elements themselves. Thus decorrelation of these signals can lead to bandwidth compression without significantly affecting image resolution. Moreover compression techniques exploit the insensitivity of the human visual system of certain spatio-temporal visual information. Main standard video codecs are based on two fundamental redundancy reduction principles:

- Spatial redundancy reduction: compression of similar pixels within the frames.
- Temporal redundancy reduction: to remove similarities between the successive pictures

2.2.1 Video Structure

As shown in Figure 4 [8], a standard video structure is built as follows:

1) *Group of pictures (GOP)* A GOP is a series of pictures. The first coded picture in the group is an I-frame, which is followed by an arrangement of P- and B-frames.
2) *Picture* Standard video codecs defines three types of pictures:
   - *Intra Frame*: I-frames are reference pictures and are coded using only information present in the picture itself (spatial compression). They are thus independant of other frames.
   - *Predicted Frame*: P-frames are coded with respect to the nearest previous I- or P-frame. This technique is called forward prediction. Moreover P-frames use motion compensation (cf. Section 2.2.3) to provide more compression than is possible with I-frames (temporal compression).
   - *Bidirectional Frame*: B-frames use both a past and future picture as a reference. This technique called bidirectional prediction provides the most compression but also the largest computation time.

   A picture consists of three rectangular matrices representing luminance (Y) and two undersampled chrominance (Cb and Cr) values [1].
3) *Slice* A slice is a group of macroblocks. The reason for defining a slice is to prevent channel error propagation into the picture by skipping to the next slice.

1. YUV422 pixel format provides 2 Bytes per pixel [1][2]
2. b=bit and B=Byte
3. Note that a new wireless protocol is being developed with 540Mbps performance: *IEEE 802.11n*

4) *Macroblock* A macroblock refers to a 16x16 block of pixels in a picture. In the YUV4:2:0 colorspace [2], a macroblock is represented as six 8x8 blocks: 4 of these blocks are located in the Y plane and 1 in each Cb and Cr plane.
5) *Block* It is the smallest coding unit and consists of 8x8 pixels for both luminance and chrominance components.

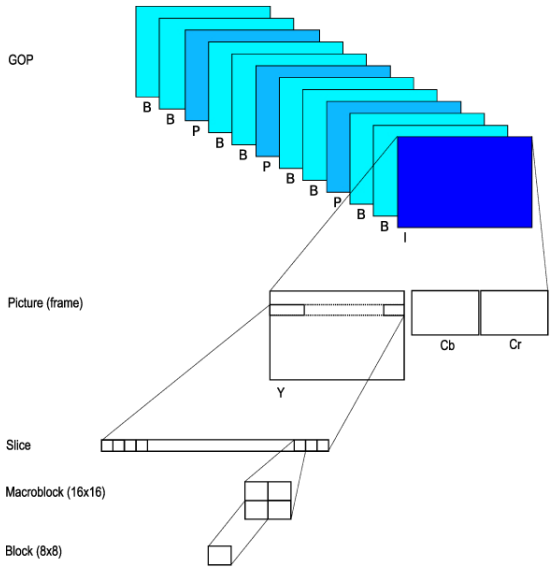

Fig. 4. Codec Video Structure

2.2.2 Spatial Compression

In video coding, main spatial compressions are based on JPEG standard [9]. For YUV pictures, spatial compression is carried out in 4 stages[4]. Figure 5 presents the path to achieve image compression.

1) *Block extraction*
Each channel of a frame is divided into 8x8 blocks in order to decrease the number of operation. A block is considered as a unit of video compression.
2) *Discrete Cosine Transform (DCT)*
A DCT consists of converting each block from spatial domain to frequency space. The DCT algorithm is built in order to separate low and high frequency (low at the top-left corner of the block and high at the bottom-right). However eyes sensitivity to spatial-temporal pattern decreases with high spatial and temporal frequency [11]. Thus the force of DCT is to exploit this human property directly by gathering most of the signal in the corner of low frequency. The next step of quantization will filter high frequency without deteriorating quality too much. Discrete Cosine Transform formula is given by:

$$\mathbf{DCT}(i,j) = \frac{2}{N} C(i)C(j) \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} pixel(x,y) \cos[\frac{(2x+1)i\pi}{2N}] cos[\frac{(2y+1)j\pi}{2N}] \quad (1)$$

$$C(x) = \begin{cases} \frac{1}{\sqrt{2}} & \text{for } x = 0 \\ 1 & \text{for } x > 0 \end{cases}$$

3) *Quantization*
This step reduces the amount of information in the high frequency components by simply dividing DCT matrix by a quantization matrix carefully built, and then rounding the components to the nearest integer. This is the main lossy operation in the whole process. As a result of this, it is typically the case that many of the higher frequency components are rounded to zero and many of the rest become small positive or negative numbers, which take many fewer bits to store. However, if the quantification is too large (large compression rate), there won't be enough coefficients to represent the block and then this one will appear visible and the frame pixellized. A common quantization matrix Q is given by:

$$Q(i,j) = 1 + (1 + i + j) * q \quad (2)$$

, where q is a quality factor[5]. For example q=5 gives:

$$Q = \begin{pmatrix} 6 & 11 & 16 & 21 & 26 & 31 & 36 & 41 \\ 11 & 16 & 21 & 26 & 31 & 36 & 41 & 46 \\ 16 & 21 & 26 & 31 & 36 & 41 & 46 & 51 \\ 21 & 26 & 31 & 36 & 41 & 46 & 51 & 56 \\ 26 & 31 & 36 & 41 & 46 & 51 & 56 & 61 \\ 31 & 36 & 41 & 46 & 51 & 56 & 61 & 66 \\ 36 & 41 & 46 & 51 & 56 & 61 & 66 & 71 \\ 41 & 46 & 51 & 56 & 61 & 66 & 71 & 76 \end{pmatrix}$$

4) *Entropy coding*
Entropy coding is a special form of lossless data compression. At first coefficients are organized in a zigzag order to produce long runs of zero. Then a run-length encoding (RLE) algorithm is applied on zero coefficients. These ones are replaced with their consecutively appearance frequency. For example {1 0 0 0 0 0 0 0 -2 } provides {1 #7 -2}. The last zero sequence is replaced with the symbol EOB. This mechanism strongly reduces block information. Finally Huffman encoding, also based on occurrence frequency, provides bit sequences for each remaining coefficients.

Decoding to display the image consists of doing all the above in reverse. Only the quantization step results in data loss. It is the cause of artifacts which can be reduced by choosing a lower level of compression.

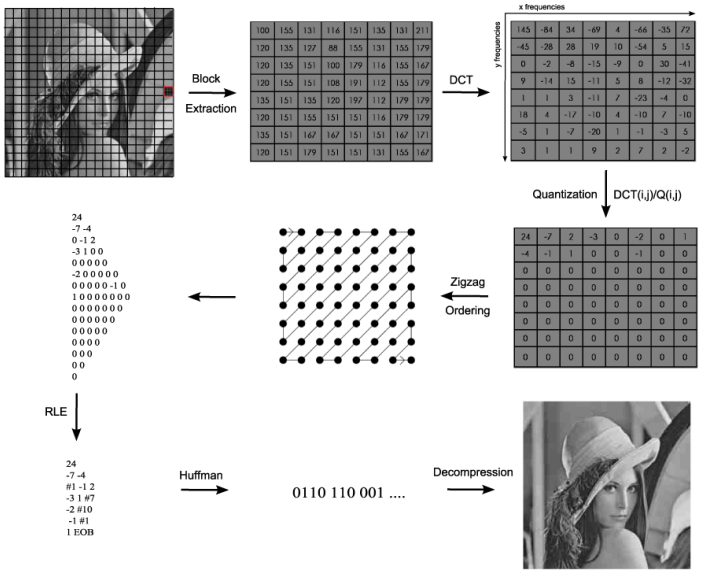

Fig. 5. Spatial compression mechanism

4. For RGB pictures a color space transformation is carried out from RGB to YCbCr space

5. q is the common compression rate which can be adjusted in an image processing software as Photoshop

2.2.3 Temporal Compression

Temporal compression applies to predictive (P) and bidirectional (B) frames using motion compensation based prediction. General concept of this principle is to search for each macroblock of the current frame which macroblock of the previous I- or P- frame closely matches the macroblock in consideration. This results in a two-dimensional motion vector and strongly reduces data. The difference between the two macroblocks provides the prediction error, which is compressed in the same way as spatial compression (DCT, quantization, RLE and Huffman). The processing of B-frames is similar except that B-frames use the picture in the following reference as well as the picture in the preceding reference frame. As shown in Figure 6, the coder provides, for a frame, motion vectors and prediction errors for each macroblocks. Thus, the decoder just has to mix the information to reconstruct the picture.

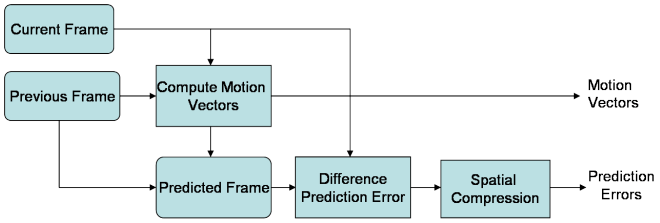

Fig. 6. Temporal compression mechanism

2.3 Codec Implementation Considerations

Among all existing codecs, the codec selected for this project has to answer several criteria:

- It has to work in *real time*. It is a necessary condition to emulate reality through HMD. The video acquired from the camera must be displayed as soon as possible on glasses to avoid delay annoying for users.
- The codec must allow frame rate and compression rate parameters to vary in real time. This constraint is required to follow network bandwidth variations.
- It must be possible to *implement* it in a software. For further applications it could be interesting to work with embed code or to use the codec on different platform.

This section exposes the system implementation structure of this project to achieve video acquisition and compression.

2.3.1 System Implementation Structure

The system implementation structure presents three main blocks (Figure 7):

1) *Video Acquisition and display* (SeeSaw Block)
   Within the framework of this project, video acquisition on MacOSX platform[6] is based on the seeSaw example application by Daniel Heckenberg[13]. As explained in his paper[14], seeSaw code takes advantage of QuickTime Video Digitizer Components (Vdig) acquisition and OpenGL image display to achieve high performance, low latency image processing. Thus this code is appropriated for our project which needs low latency to provide nice interaction with user's environment.

6. For cross-platform implementation, portVideo[12] is a framework that provides uniform access to camera device for video processing or display.

2) *FFmpeg framework*
   FFmpeg[15] is a free cross-platform multimedia framework that can record, convert and stream digital audio and video. Many open source applications rely on it such as *VLC Media Player*, *MPlayer* or *FFmpegX* on Mac OS X. The part used for this project is *libavcodec*, which is a library supporting most of existing codecs.
3) *Codec implementation*
   Codec implementation is carried out with FFmpeg. Codec implementation compresses video stream acquired by the QuickTime Video Digitizer Components (Vdig) and then provides the decompressed video stream to OpenGL which displays it. This block is in the middle of the implementation because it links the video acquisition and display achieved by seeSaw with the video compression and decompression based on FFmpeg.

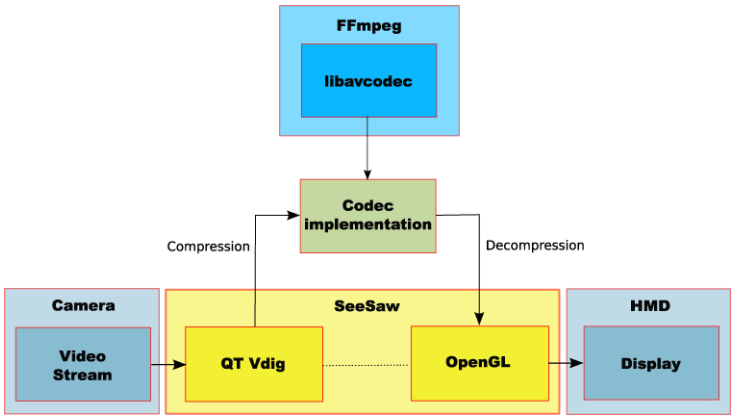

Fig. 7. System Implementation Structure

2.3.2 Codec Selection

Once video acquisition and codec implementation are defined, the codec that matches the predefined criteria has to be selected. Main codecs presented in section 2.1 were thus implemented. It appears that MPEG-4 standard best satisfies the conditions. Indeed MPEG-1 and MPEG-2 are rejected because they offer less compression than MPEG-4. In the same way, the ITU standards (H.261 ->H.264) are not usable with 640x480 video resolution[7]. Thus, to use them, it is necessary to convert the frame resolution. This process is time consuming and could increase the latency.
So the codec's choice focuses on MPEG-4 standard, which provides several advantages. Not only this codec is able to work in real time with all parameters values, but it is also able to follow low to high bit rates. In addition, it has already prove reliable in videoconference.

2.4 Codec System Identification

The codec, seen as a system, provides a bit rate[8] according to the encoding parameters configurations. Figure 8 illustrates this concept. But the bit rate also depends on video content. Indeed, for codecs based on predictive frames (P-frames) or bidirectionnal frames (B-frames), where the difference between

7. Valid resolutions for H.26x are 128x96, 176x144, 352x288, 704x576 and 1408x1152

8. the concerned bit rate is that between compression and decompression (transmission in Figure 1).

two successive frames is coded, high motion in the video sequence strongly increases this difference. Thus, codec system provides a higher bit rate than with low motion. In the same way, if the frames of the video sequence contains a lot of details, video encoding will require more information than for *poor* pictures. Hence, the bit rate depends on spatial and temporal video content. Consequently, codec system must be identified according to its parameters and video content to know its properties and the bit rate behavior.

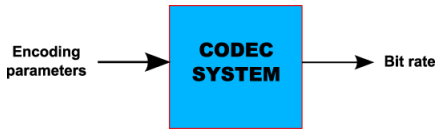

Fig. 8. Codec System

For this project, only two encoding parameters are selected to simplify the parameters space: the *frame rate* and the *compression rate*. The frame rate and compression are chosen because they have more impact on user perception than other parameters such as resolution or color depth. To take video content into account, bit rate characteristics are plotted for three cases:

- *Static*: The camera is fixed, there is no motion in the video content.
- *Low Dynamic*: The camera is moved slowly, there is low motion in the video content.
- *High Dynamic*: The camera is moved quickly, there is high motion in the video content.

For the two last cases, spatial content varies from low to high details in order to cover all content space. Codec system is identified for this configuration:

- *Codec*: MPEG-4 Part 2 with a key frame (I-frame) every ten frames (gop size=10) and no B-frame. Including a certain gop size permits to achieve predictive coding which decreases the bit rate. For example, low compression rate video sequence at 30fps provides 21Mbps with I-frames only and 7Mbps with a gop size of ten. Including B-frames increases encoding time because B-frames depend on future frames and codec has to decode the future frames before decoding the B-frames. This should add some latency, which would be harmful for user perception.
- *Camera*: *iSight* webcam delivering 640x480 resolution at 30fps video stream.
- *Computer*: *PowerMac* with *2x2 GHz PowerPC G5* processor, *1Go DDR SDRAM* Memory and *ATI Radeon 9600 Pro* graphics card.

2.4.1 Parameters specifications

The frame rate can vary on a continuous scale from 1 to 30fps and the compression rate on a discrete scale from 0 to 32'767. The compression rate is a FFmpeg intrinsic parameter. It depends on other parameters and acts on the quality factor of spatial compression. So it doesn't really corresponds to a ratio and is henceforth called *quality parameter*. For this scale, 0 is the best quality and 32'767 the worst.

2.4.2 Bit rate characteristic according to frame rate

The bit rate is computed for a gop size (ten frames) to take account that I-frames have more information than P-frames. For example, with a gop size of ten, 1002 quality video sequence at 10fps provides a 23kbits I-frame and 1kbits P-frames. High gop size would not be practical because at low frame rates computing the bit rate would take a lot of time. A gop size of ten frames seems to be a good compromise. Figure 9 shows the bit rate behavior for a quality parameter of 500, for the three content types and for nine frame rate values: {1, 5, 7, 10, 12, 15, 20, 25, 30}. Figure 10 presents the difference[9] between the global average bit rate and the maximum bit rate.

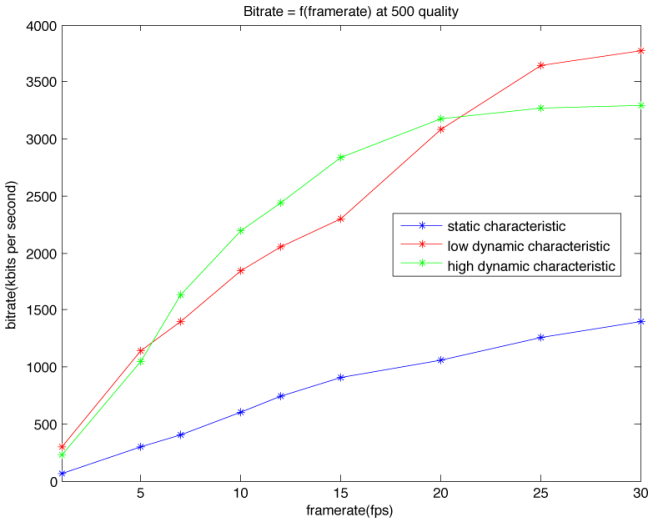

Fig. 9. Characteristic bitrate=f(framerate) at 500 quality parameter

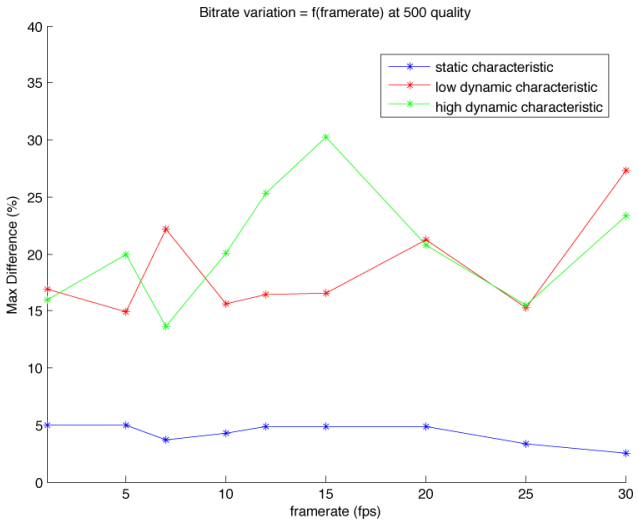

Fig. 10. Characteristic bitrate variation=f(framerate) at 500 quality parameter

As expected, the static characteristic is linear and stable (maximum variation $< 5\%$). An increase of the frame rate corresponds to a proportional increase of the bit rate.
The low and high dynamic characteristics are very close. They logically corresponds to higher bit rates than static characteristics. This is due to the fact that P-frames have more data due to the motion of the camera. Then the average maximum variations are also higher (approximately 18% and 22% respectively). This can be explained by the fact that there are

9. in percent of the global average bit rate

great fluctuations of spatial video content, which is not the case with a static camera. The characteristics tend to be flattened at high frame rates. Indeed, for a constant motion, low frame rates induce few correlation between frames. The difference between two successive frames is then high. This drives to high information P-frames. On the contrary, high frame rates provide low difference between two consecutive frame (two consecutive frames have almost same information). This results in low information P-frames and lower bit rate than expected. This explains non linearities which appear with motion.

However the three characteristics can be considered as linear, knowing that modeling error could be compensated by a controller. Other quality values were tested and showed similar trends.

2.4.3 *Bit rate characteristic according to quality parameter*

The bit rate is computed in the same way as before. Figure 11 shows the bit rate behavior for 15 frames per second, for the three content types and for eleven quality values: {0, 200, 400, 600, 800, 1000, 1500, 2000, 3000, 5000, 10'000}.

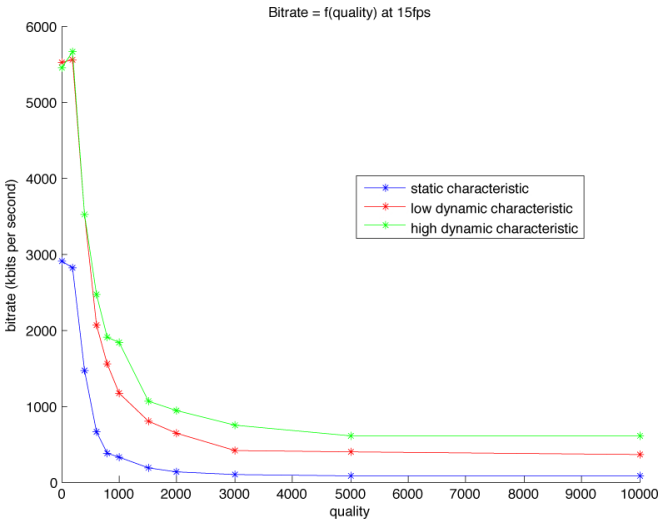

Fig. 11. Characteristic bitrate=f(quality) at 15 fps

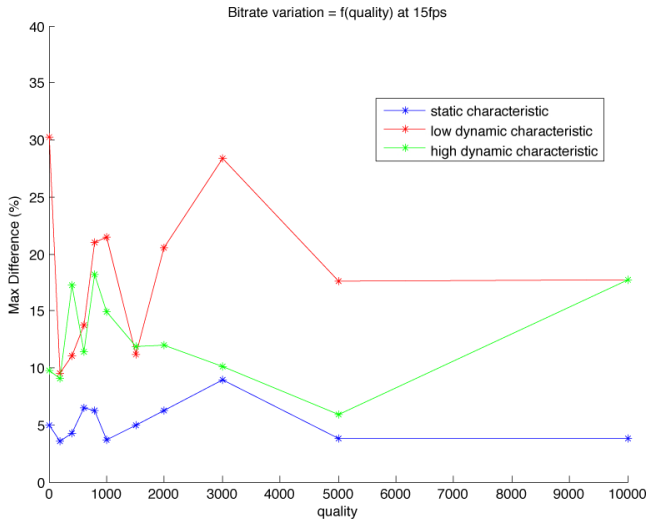

Fig. 12. Characteristic bitrate variation= f(quality) at 15 fps

The characteristics are exponential for the considered cases. This cannot be intuitively explained because the characteristics depend on how FFmpeg achieve video encoding and decoding. High and low dynamic characteristics are very close. For the same reason as in 2.4.3, the static case provides lower bit rates. However it can be noted that some bit rate values of a same characteristic are curiously close. It is the case for the two first qualities: 0 and 200. This observation could reflect the presence of stages for which the bit rate remains constant. In order to verify this assumption, a static experiment at 30 fps is undertaken. The bandwidth is evaluated for each quality. These tests clearly showed the presence of thirty thresholds which significantly modify the bit rate, as shown in Figure 13. Thus system identification shows that the quality parameter can vary on a discrete scale of thirty values: {0, 295, 413, 531, 649, 767, 885, 1002, 1120, 1238, 1356, 1474, 1592, 1710, 1827, 1945, 2063, 2181, 2299, 2417, 2535, 2653, 2770, 2888, 3006, 3124, 3242, 3360, 3478, 3596}.

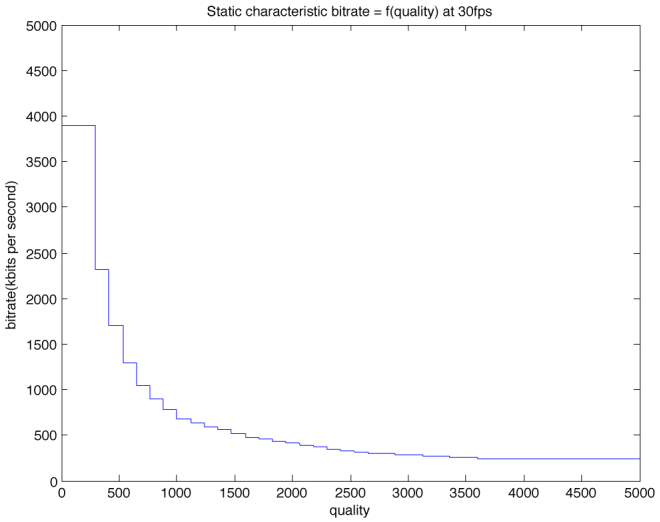

Fig. 13. Static characteristic bitrate= f(quality) at 30 fps

2.4.4 *Video coding delay characteristic*

This section shows the effect of both encoding parameters on the video coding delay. Video coding delay is the time that needs the codec to encode and decode a frame. The setup configuration is the same as presented in section 2.4. Figure 14 and Figure 15 shows the delay characteristic according to frame rate respectively quality for a high dynamic case.

It can be noted that delay decreases when frame rate increases. This is due to the fact that high frame rates provide less difference between two successive frames than low frame rates. Thus less information have to be coded at high frame rates and encoding time decreases. In the same way delay decreases when quality increases[10]. For high quality values, lot of information are lost when encoding the video frames. Hence, the encoding time is low.

The goal of these characteristics is to learn the impact of video coding delay on user perception. The threshold of human perceived latency is about 50ms [14]. Given that the frame duration of a broadcast standard based video device is at least 33ms (for 30fps) and that video input device has two

10. Note that an increase of the quality parameter corresponds to a decrease of the perceived-quality.

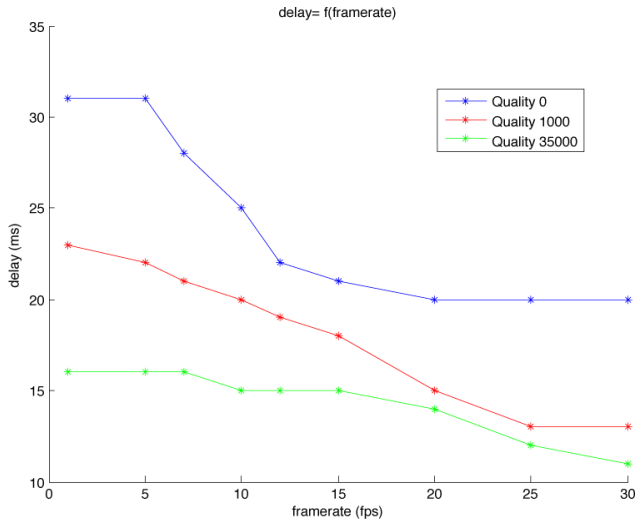

Fig. 14. Dynamic characteristic delay= f(framerate)

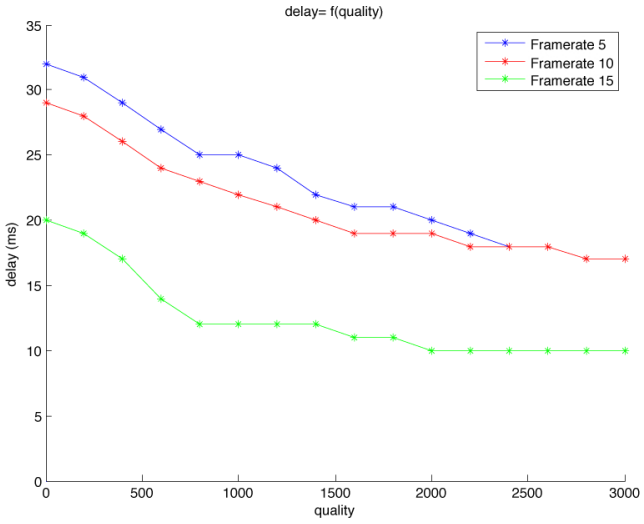

Fig. 15. Dynamic characteristic delay= f(quality)

frames of latency [14], video coding delay could be harmful for user perception. Experiments were achieved to find a threshold where video coding delay becomes annoying for user perception. Unfortunately, no threshold was found due to some unexplainable observations. For example the pairs (25fps, 0 quality) and (21fps, 1000 quality) both correspond to 20 ms coding delay but the latency perception is completely different. For the first one, latency is very annoying and for the second latency is not perceived. Another example, 30 fps is very annoying and provides greater latency than lower frame rates. That is not intuitive because delay characteristics show that an increase of the frame rate tends to decrease the delay. It has also been observed that quality 0 always provides very annoying latency. The same tests were achieved on a G4 Macintosh platform. All delays were bounded between 21 ms and 40 ms and annoying latency was always present. So, tests strongly depends on the configuration setup.

3 Perception Model

Once the encoding system is identified, a perception model is built to provide the best encoding parameters in response to a given network bit rate (cf. Figure 16). This model must be able to provide via a *Local Adaptation Scheme* (LAS), for each bit rate value and within the set of different means to achieve this target bit rate, the best encoding configuration that maximizes the user-perceived quality. Thus, the perception model is an Optimal Adaptation Path (OAP) in the encoding parameters space (or adaptation space). If a particular system has n independant parameters that define the encoding configuration, then there exists an adaptation space with n dimensions where each dimension represents an independant encoding configuration. It results in an n dimensions OAP.

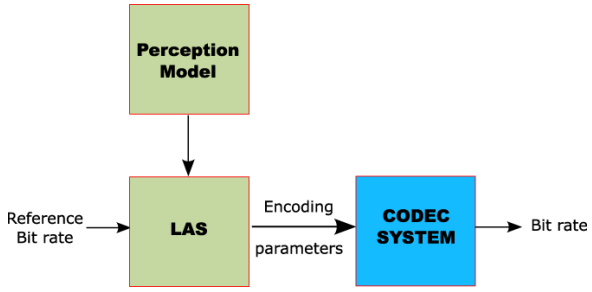

Fig. 16. Open Loop system with perception model

In the 6^{th} *Sense* project, a distinction is made between Global Optimal Adaptation Path (GOAP) and Local Optimal Adaptation Path (LOAP). The first one refers to how to share network bandwidth among streams (audio, video) in an optimal way, whereas the second one adapts encoding parameters for a bit rate given by the GOAP to maximize the user-perceived quality. *Virtual See-Through* project only focuses on the LOAP, whereas GOAP will be explored in future work.
This section proposes a method to discover the Local Optimal Adaptation Path. Concepts adaptation space and LOAP are first exposed. Then several methods of video quality evaluation are explored. Finally, the evaluation process for the selected methodology is presented and a LOAP is proposed.

3.1 Adaptation Space and Local Optimal Adaptation Path Concepts

The work presented here focuses on the adaptation of MPEG-4 video streams within a two-dimensional adaptation space defined by frame rate and quality parameters. Frame rate axis is a continue axis of frame rate values between 1 and 30 fps. On the other hand, codec identification show that quality axis contains thirty discrete quality values from 0 to 3596. For each possible pair of parameters correspond a certain bit rate. As mentionned above, this bit rate strongly depends on the spatial and temporal content of the video. To be completely rigorous, each content should correspond an adaptation space with different bit rates. Nicola Cranley [16] proposes a method to take into account the video content in the adaptation process. However, in this project only one adaptation space is considered to standardize the perception model for all type of content and to avoid additonal processing which could add some latency which is harmful for video perception. Finally, in order to know the bit rate value for a given encoding parameters pair in the parameters space, some basic[11] bit rates have to be fixed. To achieve that, the bit rates

11. The term *basic* is selected to avoid any confusion with the reference bit rate

for each quality at 15 fps and with a low dynamic content have been calculated (cf. first and last columns of Table 1). This configuration has been selected according to the applications of the project: chemical plant interaction with slow movements. Thus, each *bitrate* corresponding to a given pair *(fr, q)* may be found from a $bitrate_{basic}$ value, assuming that the bit rate characteristic according to the frame rate is linear:

$$bitrate(fr, q) = \frac{fr}{15} bitrate_{basic}(15, q) \quad (3)$$

For example, the encoding point (20_{fr},1710_q) corresponds to 721 kbps bit rate and (11.3_{fr},2888_q) to 243 kbps.

To permit an intuitive reading of the adaptation space, all encoding parameters should be represented in an equivalent way. However this is not the case with the quality scale. Indeed, an increase of the quality corresponds to a decrease of the bit rate. In addition, the codec system provide a non-linear relation between quality and bit rate, which makes even more difficult the interpretation of the parameters space. Consequently, we propose to redefine the quality scale in order to provide an easier interpretation of the adaptation space. The goal of this redefinition is to have a correlation between the two axis in terms of their effect on the system output: the bit rate. The characteristic of the bit rate according to the two parameters must be linear in the parameters space. Hence, for a given parameters set, to double the frame rate, value only shall have the same effect as to double the quality value only. Thus, this representation offers a more intuitive reading of the adaptation space. The new quality scale is expressed in terms of procent of the maximum basic bit rate value:

$$quality_{procent} = \frac{bitrate_{basic}(15, quality)}{4700} 100 \quad (4)$$

Table 1 presents the new discrete quality values. Now an increase of the quality logically corresponds to an increase of the bit rate and their relation is linear.

Due to the discrete nature of the quality parameter, values between the quality thresholds are not reachable in the parameters space. On the other hand, for a given quality, one may travel along all frame rate axis. Figure 17 shows the new adaptation space and the possible encoding points in blue.

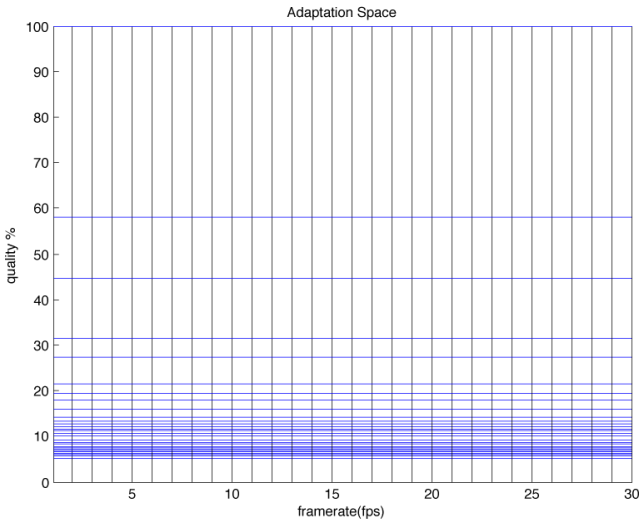

Fig. 17. Adaptation space of possible encoding couples

Quality	$Quality_{procent}$ %	$Bitrate_{basic}$ kbps
0	100	4700
295	58.02	2727
413	44.44	2089
531	31.38	1475
649	27.42	1289
767	21.59	1015
885	19.46	915
1002	17.89	841
1120	15.85	745
1238	14.10	663
1356	13.38	629
1474	12.76	600
1592	12.25	576
1710	11.51	541
1827	11.25	529
1945	10.59	498
2063	9.95	468
2181	9.27	436
2299	8.65	407
2417	8.40	395
2535	7.85	369
2653	7.55	355
2770	7.10	334
2888	6.85	322
3006	6.57	309
3124	6.23	293
3242	6.04	284
3360	5.68	267
3478	5.57	262
3596	5.08	239

TABLE 1
Basic Bit rate at 15fps

It appears that for a given bit rate, different encoding configurations may correspond. For example a bit rate of 3200kbps provides three possible pairs: (10.21_{fr} , 100_q), (17.6_{fr} , 58.02_q) and (22.97_{fr} , 44.44_q). This introduces the concept of isobitrate, which provides all encoding configurations for a given bit rate. Figure 18 shows the shape of the isobitrate at 400kbps.

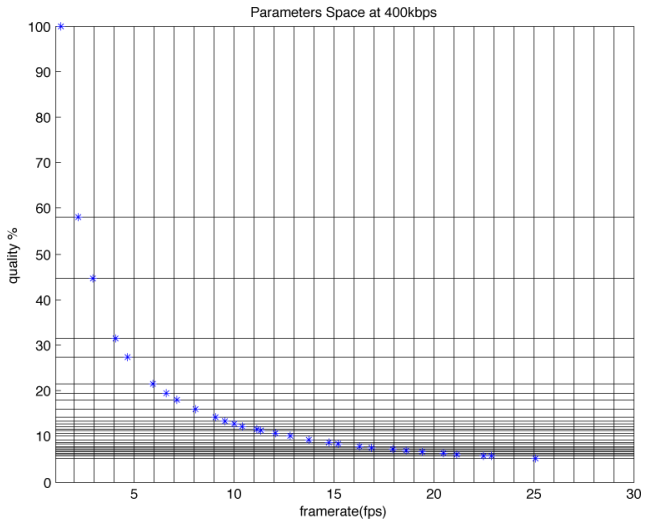

Fig. 18. Adaptation Space at 400kbps isobitrate

Finally, the work of the LOAP is to supply for each bit rate of the adaptation space the best encoding configuration which maximize the user-perceived quality. This raises the question of how user-perceived quality can be assessed in practice. It is

the object of the next section.

3.2 Video Quality Evaluation

Many methodologies are proposed in order to compare different video qualities. It exists two types of quality evaluation: *objective* and *subjective*. Subjective video quality evaluation is concerned with how video is perceived by a viewer. This method strongly approaches the human visual system (HVS) but is time consuming because several tests with different users are necessary. Objective video quality evaluation techniques are mathematical models that emulate the subjective quality assessment results, based on criteria and metrics that can be measured objectively. Objective metrics provide an immediate evaluation but generally don't match well to the characteristics of the human visual system.

3.2.1 Objective Metrics

The objectivity of the methods is owed to the fact that there is no human interaction; the original video sequence and the impaired one (the compressed video) are fed to a computer algorithm that calculates the distortion between the two (Figure 19).

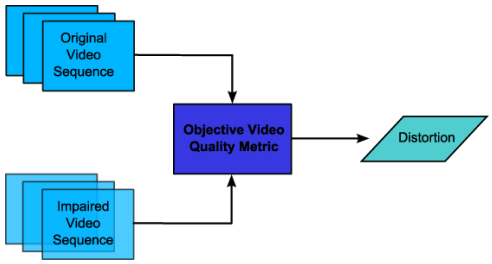

Fig. 19. Objective metric mechanism

Some basic objective metrics are presented her:

- *Mean-Square Error (MSE)*
  The mean squared error is the most popular difference metric in image and video processing with the peak signal-to-noise ratio (PSNR). The MSE is the mean of the squared differences between the gray-level values of pixels in two pictures or sequences I and $\tilde{I}$:

$$MSE = \frac{1}{TXY} \sum_t \sum_x \sum_y [I(t,x,y) - \tilde{I}(t,x,y)]^2 \quad (5)$$

  for pictures of size $X \times Y$ and T frames in the sequence. The average difference per pixel is thus given by the root mean squared error $RMSE = \sqrt{MSE}$.

- *Peak-Signal-to-Noise Ratio (PSNR)*
  The PSNR in decibels is defined as:

$$PSNR = 10log\frac{m^2}{MSE} \quad (6)$$

  where m is the maximum value that a pixel can take (e.g. 255 for 8-bit images). PSNR, like MSE, is well-defined only for luminance information: once color comes into play, there is no agreement on the computation of these measures.

Technically, MSE measures image difference, whereas PSNR measures image fidelity, i.e. how closely an image resembles a reference image, usually the uncorrupted original. The popularity of these two metrics is rooted in the fact that computing MSE ans PSNR is very easy and fast. Because they are based on pixel-by-pixel comparison of images, however, they only have a limited, approximate relationship with the distortion or quality perceived by the human visual system. For example, the MSE can be produced in a number of different ways. That is, consider an image where the pixel values have been altered slightly over the entire image and an image where there is a concentrated alteration in a small part of the image, both will result in the same MSE value but one will be more perceptible to the user than the other. Another example, the PSNR metric does not take the visual masking phenomenon[12] into consideration, i.e. every single errored pixel contributes to the decrease of the PSNR, even if this error is not perceived. These problems prompted the itensified study of vision models and visual quality metrics in recent years. New approaches based on HVS-models and approved by the Video Quality Experts Group (VQEG) [19] are slowly replacing classical schemes. Among them we can find:

- *Perceptual Distortion Metric (PDM)*
  The perceptual distortion metric (PDM) developed by S. Winkler [18] is based on a spatio-temporal model of the human visual system. It consists of four stages processing both the reference and the degraded sequence (cf. Figure 20). The first stage converts the input to an opponent colour space, which states that the color information received by the cones is encoded as white-black (W-B), red-green (R-G) and blue-yellow (B-Y) color difference signals. The second stage implements a spatio-temporal perceptual decomposition into separate visual perception channels of different temporal frequency, spatial frequency and orientation. The third stage weights each channel according to spatio-temporal contrast sensitivity. The final stage of the metric is a detection stage that computes a distortion measurement from the difference between the measured parameters of the reference and degraded clips.

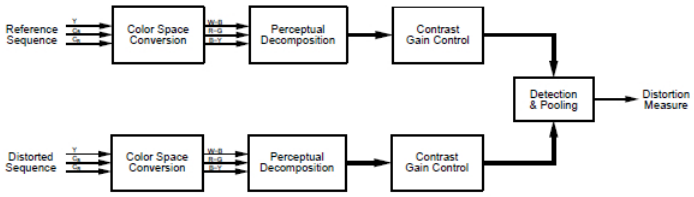

Fig. 20. Block diagram of the perceptual distortion metric (PDM)

A lot of other quality metrics based on HVS-models have been developped such as *Video Quality Metric (VQM)* [20], *Perceptual Video Quality Metric (PVQM)* [21] or *Motion Picture Quality Metric (MPQM)* [22].

3.2.2 Subjective Test Methodologies

Subjective testing for visual quality assesment has been formalized in ITU-R Recommendation BT.500-11(2002) [23], which suggests standard viewing conditions, criteria for the selection of observers and test material, assessment procedures, and data

12. Masking is a HVS property which occurs when a stimulus that is visible by itself cannot be detected due to the presence of another [18]

analysis methods. The three most commonly used procedures are the following:

- *Double Stimulus Continuous Quality Scale (DSCQS)*
  The presentation sequence for a DSCQS trial is illustrated in Figure 21. Viewers are shown multiple sequence pairs consisting of a *reference* and a *test* sequence, which are rather short (typically 10 seconds). The reference and test sequence are presented twice in alternating fashion, with the order of the two chosen randomly for each trial. Subjects are not informed which is the reference and which is the test sequence. They rate each of the two separately on a continuous quality scale ranging from *bad* to *excellent* as shown in Figure 22. Analysis is based on the difference in rating for each pair, which is calculated from an equivalent numerical scale from 0 to 100. This differencing removes a lot of subjectivity with respect to scene content and experience.
- *Double Stimulus Impairment Scale (DSIS)*
  The presentation sequence for a DSIS trial is illustrated in Figure 23. As opposed to the DSCQS method, the reference is always shown before the test sequence, and neither is repeated. Subjects rate the amount of impairment in the test sequence on a discrete five-level scale ranging from *very annoying* to *imperceptible* as shown in Figure 24.
- *Single Stimulus Continuous Quality Evaluation (SSCQE)*
  Instead of seeing separate short sequence pairs, viewers watch a program of typically 20-30 minutes duration which has been processed by the system under test; the reference is not shown. Using a slider, the subjects continuously rate the instantaneously perceived quality on the DSCQS scale from *bad* to *excellent*.

Another subjective methodology, which is often employed in cognitive science is the:

- *Forced Choice Methodology*
  In forced choice, the subject is presented with a pair of alternatives separated by a short gap or signal. The subject must choose one of the alternatives according to some test criteria (cf. Figure 25). At the beginning of the test procedure, the reference clip is shown. During a single trial, the subject is shown two degraded versions of the same clip, A and B. A degraded version of the clip is one with a lower encoding configuration. These clips are shown consecutively separated by a short signal or gap. The subjects task is to choose whether the first or second clip was better. In forced choice, there are equally probable alternative degraded versions of the clip between which the subject must choose. When a subject cannot make a decision, they are forced to make a choice.

3.2.3 Comparison of objective and subjective evaluations

Main defect of basic objective metrics is that they are not correlated with human vision, which is mainly governed by two key concepts [18]:

- *Contrast sensitivity*. The response of the human visual system depends much less on the absolute luminance than on the relation of its local variations to the surrounding luminance. Contrast sensitivity decreases with an augmentation of these spatial variations. The same appears with temporal frequencies. The faster these variations are, the more the contrast sensitivity decreases. Figure 26 shows this phenomenon.

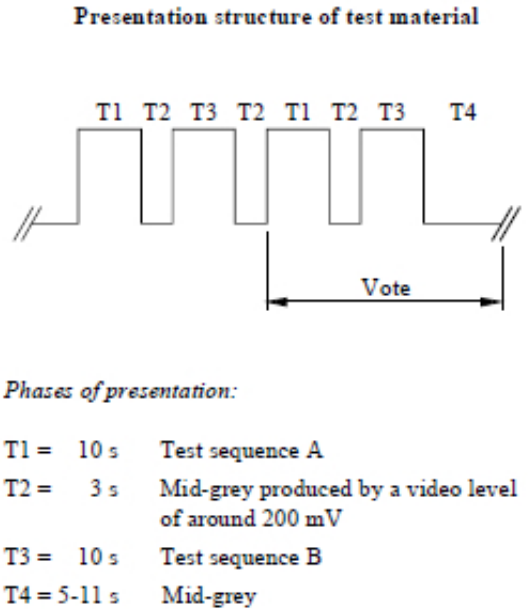

Fig. 21. Presentation structure of test material for DSCQS

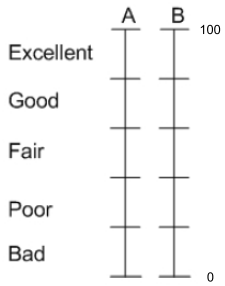

Fig. 22. Continuous quality scale

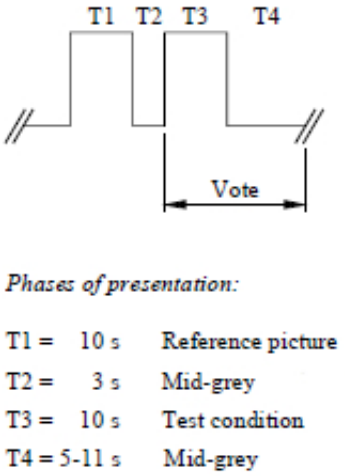

Fig. 23. Presentation structure of test material for DSIS

- *Masking*. It describes interactions between stimuli. Masking occurs when a stimulus that is visible by itself cannot be detected due to the presence of another.

	Impairment
5	Imperceptible
4	Perceptible, but not annoying
3	Slightly annoying
2	Annoying
1	Very annoying

Fig. 24. Impairment scale

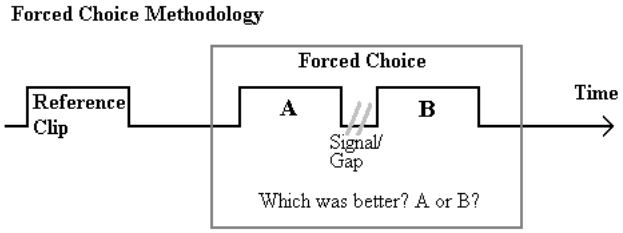

Fig. 25. Forced Choice Methodology

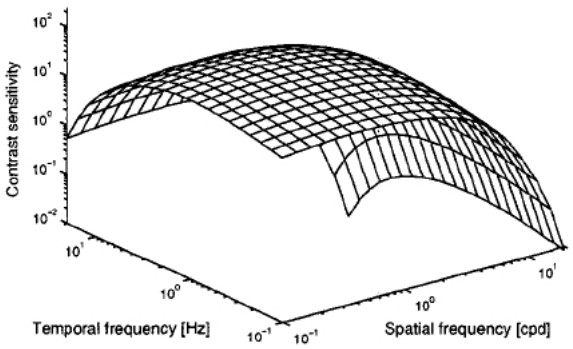

Fig. 26. Spatio-temporal contrast sensitivity [18]

Due to their pixel-by-pixel processing, basic metrics such as MSE or PSNR don't take into account these human vision properties. Thus, variations of frame rate or picture content will be perceive by an observer but the distortion processed will not be affected.
Objective metrics based on HVS-models offer a good alternative to approach the human visual system but are generally difficult to implement and less efficient than subjective testing. Subjective evaluations are without any doubt the closest we can get to the *truth* perceived quality. But testing procedures are time consuming and results analysis is generally complex. Despite everything, the ITU-T recommends that objective metrics are not a direct replacement for subjective testing. Finally, subjective quality assessment is more appropriate for research related purposes whereas objective metrics are more suited to equipment specifications and day-to-day system performance measurement.

3.3 Choice of a video quality evaluation methodology

The choice of a global video quality evaluation methodology is often restricted to the context of the project and in particular according to the applications field.

3.3.1 Virtual See-Through application considerations

In the framework of *Virtual See-Through* project, applications are directed towards the chemical industrial field. In particular, they concern the interaction between operators and chemical plants. In this context quality video evaluation differ from traditional quality assessments, where observers *passively* grade quality from a video sequence. In our case, subjects become *active* and are requested to evaluate their perception of a certain interaction with the plant through the Human Mounted Display (HMD). Thus, perception is not only evaluated in term of video quality, as it is in general the case, but rather in term of *feeling* or *quality of perception (QoP)* with a given interaction. Some representative operations were selected from industrial procedures on chemical plant:

- Read a level gauge of liquid
- Engage vacuum pumps by pressure of a button
- Opening of the manual vapor valve
- Control pressure of a liquid using a barometer
- Close the cooling water valves

These interaction types specified the context in which perception is evaluated. Hence, there are mainly three factors that can affect user's quality of perception:

1) The *fluidity* of video content: it is given by the frame rate parameter.
2) The *quality* of video pictures: it is given by the quality parameter. The quality is defined by the level of details.
3) The *latency* between pictures acquisition and display. This factor doesn't appear in classical video quality evaluations because it has no influence on *passive* perception. On the other hand, latency is determining with interactions. It can have a very annoying effect on *active* perception. For example, a delay of one second between a valve opening and the real vision of this operation is annoying for the perception.

Thus, the selection of a video quality evaluation methodology should reflect the goal of the experiments and should take these factors into account.

3.3.2 Choice of a video quality evaluation methodology

As explained in section 3.2, subjective evaluations give better results than objective metrics. On this subject, Nicola Cranley compares in [17] the results obtained using subjective means with results obtained using several sophisticated objective metrics and concludes that objective metrics for adapting video quality are not satisfactory in quantifying human perception. Moreover, objective metrics, even those based on HVS-models, don't consider latency phenomenon, which is a limiting factor for this project. Consequently, objective evaluations don't satisfy project conditions and the choice will be done among the subjective evaluations. Despite everything, improving actual objective metrics can be the subject of future work. Subjective testing is then necessary to evaluate new metric quality.
The most suitable subjective methodology, presented in section 3.2.2, is the *Forced Choice Methodology*. The choice of Forced Choice Methodology is motivated by several factors:

- The simplicity of the grading scale. Indeed, a known criticism of the DSCQS, DSIS and SSCQE methods is the vocabulary of the impairment and quality scale. Subjects do not interpret these scales the same way and the results can be biased. This problem does not appear with Forced Choice Methodology.

- The simplicity of the statistical analysis. This is due to the simplicity of the grading scale. Analysis is more complicated with other methodologies.
- The absence of contextual effect. Contextual effects occur when the subjective rating of an image is influenced by the order and severity of impairments presented. For example, if a strongly impaired image is presented after a string of weakly impaired images, viewers may inadvertently rate this image lower than they normally might have. Some studies found that the results of the Double Stimulus Impairment Scale (DSIS) method are biased to a certain degree by contextual effects [23]. Forced Choice Methodology does not present contextual effects because the order within the pair of video sequences is randomized and does not contain the reference sequence.
- The absence of forgiveness effect. Human memory effects for quality estimation seem to be limited to about 15 seconds [24]. Thus, video clips duration and voting time must be short enough to prevent the user forgetting the quality of the previous one. However, the voting time of the Double Stimulus Continuous Quality Scale (DSCQS) generally exceeds acceptable time. With the Forced Choice Methodology, the observer take a very quick decision, choosing either the first or second clip as being better or worse.
- Non-sense of Single Stimulus Continuous Quality Evaluation (SSCQE) for this project. Indeed, SSCQE is generally used to detect change of quality in a long video sequence. This sequence has to present the same time instants for each subject. It is not the case when evaluating interaction perception. For a given interaction and a given time, user will see a different video content than another.

3.4 Finding LOAP in the Adaptation Space

A method is proposed here to find the *Local Optimal Adaptation Path* (LOAP) in the adaptation space with the Forced Choice Methodology. It is developed for this project but can be adapted to other applications. As seen in section 3.1, each bit rate correspond several encoding configurations. Thus, finding LOAP ideally consist to test all possible parameters couples of each bit rate with a subjective quality evaluation in order to find the best encoding configuration. Of course, testing all possibilities would not be feasible to a reasonable amount of time. Thus a practical method to reduce the number of tests is proposed. This one consists of four steps:

1) The first step is to sample the adaptation space in order to reduce quantity of tests. To this end, the two parameters are differently sampled:
   - Sampling of the frame rate is based on the *Weber's Law of Just Noticeable Difference (JND)* [25]. The Just Noticeable Difference is the minimum amount by which stimulus intensity must be changed in order to produce a noticeable variation in perception. Weber's Law, can be expressed as:

$$dp = k\frac{dS}{S} \tag{7}$$

     where dp is the differential change in perception, dS is the differential increase in the stimulus and S is the stimulus at the instant. In this context for any of the sensory perception, the amount of change for a present stimulus when the magnitude increases or decreases will always be perceived proportionally the same to the initial magnitude, no matter how intense the stimulus is. Integrating the above equation gives:

$$p = klnS + C \tag{8}$$

     To determine the integration constant C, put p = 0, i.e. no perception; then

$$C = -klnS_0 \tag{9}$$

     where S_0 is the stimulus threshold below which it is not perceived at all. The equation becomes by changing Stimulus S by the frame rate fr:

$$p = kln\frac{fr}{fr_0} \tag{10}$$

     Thus, with SV the number of sampled values, Max_{fr} the maximum frame rate value and Min_{fr} the minimum frame rate value, the frame rate samples are given by:

$$i = 0, 1, 2, ..., SV \quad fr(i) = fr_0 e^{\frac{i}{k}}, \tag{11}$$

$$where\ k = \frac{SV}{ln\frac{Max_{fr}}{fr_0}} \tag{12}$$

     Initially the frame rate is bounded between 5fps and 25 fps. Indeed, the HVS cannot appreciate more than 24fps and below 5fps, the video sequence should be considered as a series of images. Hence, $SV = 5$, $Max_{fr} = 25$, $Min_{fr} = 5$ and $fr_0 = 5$ provide {5, 8, 11, 17, 25} frame rate samples[13]. This defines the zone of interest. However extreme values of frame rate, i.e 1 and 30 are added in order to see their effect on user's perception. Finally, frame rate samples are given by six values:

$$\{1, 5, 8, 11, 17, 25, 30\}\ fps \tag{13}$$

   - Quality parameter sampling is achieved in experiments by evaluating effect of the thirty values on the perception. It appears that seven thresholds present a noticeable difference of perception. These values are selected as samples:

$$\{5.08, 7.85, 10.59, 14.1, 21.59, 44.44, 100\}\ \% \tag{14}$$

   Combination of the samples drives to a sampled adaptation space as seen in Figure 27. Each encoding configuration correspond to a noticeable different perception.
2) The second step is to select some bit rates of the adaptation space on which tests are carried out. The goal is to choose sufficient bit rates to cover all the parameters space but not too much in order to minimize the number of tests. Figure 28 shows the selected bit rates of *Virtual See-Through* project:

$$\{100, 200, 400, 800, 1600, 3200\}\ kbps \tag{15}$$

   Finally, the encoding configurations tested for each selected bit rate are the ones which are the closest to the sampled parameters pairs. Figure 29 shows the points of the adaptation space which will be evaluated.
3) The third step consists in defining how the encoding configurations are evaluated. The proposed methodology is to evaluate the parameters couples for each selected bit

13. the samples are rounded to the closest integer

rate with the Forced Choice Methodology. But instead of testing all possible two-by-two combinations of encoding configurations, that would be laborious, a new approach is proposed. The first pair of parameters couples which is subjected to Forced Choice is made of the two most distant bit rate couples. The worst couple is eliminated and the next pair is evaluated using again the two most distant encoding configurations. And so on until converging towards a single parameters couple. This point correspond to the best encoding configuration for the concerned bit rate. The main advantage of this method is that each observer corresponds a single optimal point for a given bit rate.

4) The last step consists in determining the best encoding configurations of each selected bit rate by calculating the average of all individual best couples. Thus, Local Optimal Adaptation Path can be interpolated through all these points of the adaptation space. Interpolation method is proposed in section 3.6.3.

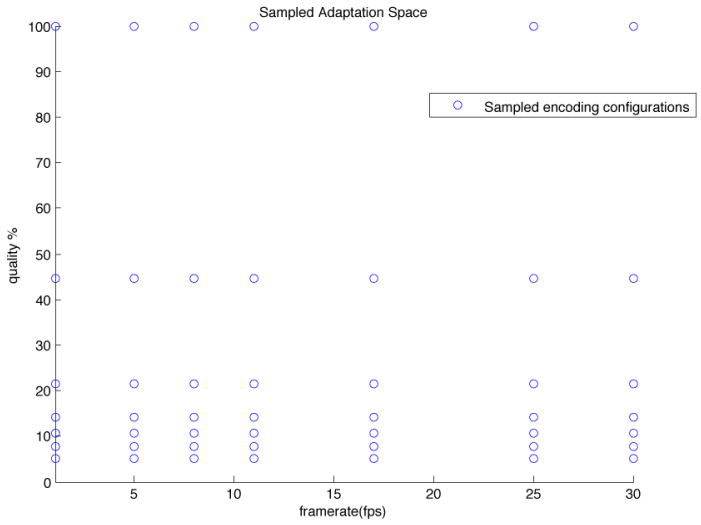

Fig. 27. Sampled Adaptation Space

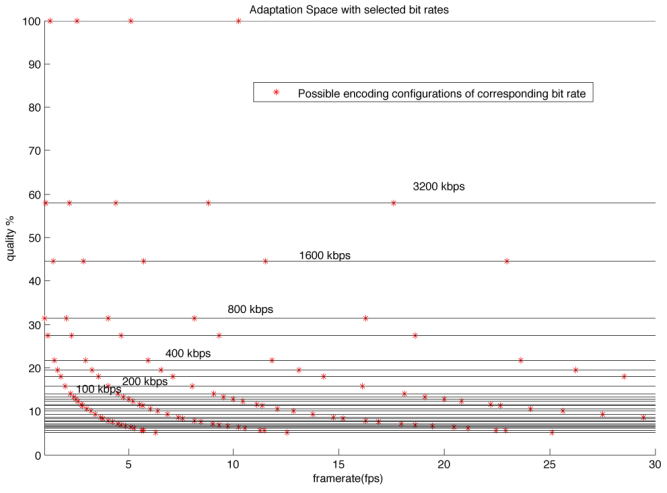

Fig. 28. Adaptation space with selected bit rates to evaluate

However, the LOAP discovering procedure is made of a series of perception evaluation tests. The tests procedure is detailed in the next section.

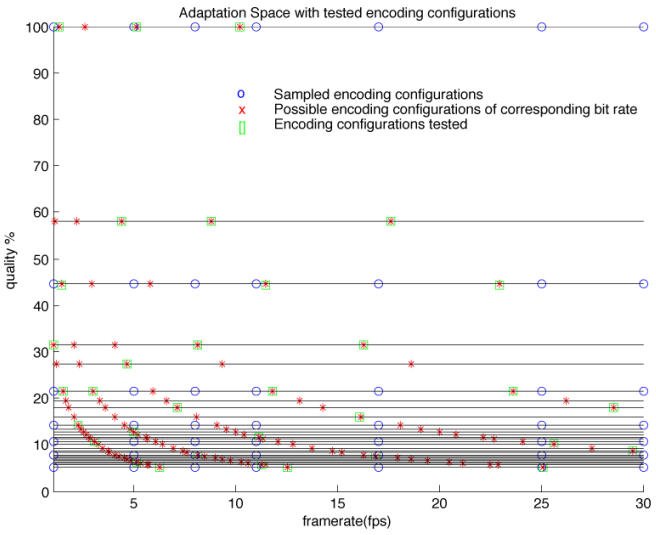

Fig. 29. Adaptation space with encoding configurations tested

3.5 Test considerations

Test procedure should reflect as well as possible the context of the real application, that is chemical process plant. All tests consist in interactions with a boiler representing a simplified chemical plant. The boiler consists of pipes, valves and a barometer and contains all the elements necessary to develop scenarios based on typical operations (cf. Section 3.3.1). In order to reflect these specific operations, two types of interaction are subjected to evaluation:

- Passive interaction, named scenario A. This operation simply consists in reading a label of the boiler. It is a *static* operation.
- Active interaction, named scenario B. Subjects are requested to open a valve of decompression, to open a purging valve and to fill a recipient with water, then to close the purging valve and finally to close the first opening valve. The operation is *dynamic*. Several manipulations in scenario B permit the subject not to familiarize too much with the scenario. Indeed, if only one operation were selected, user became familiar with it and the evaluation of perception would not take into account the interaction itself.

Results strongly depend on hardware. For this project, the hardware configuration is the same as in section 2.4:

- *Codec*: MPEG-4 Part 2 with a key frame (I-frame) every ten frames (gop size=10) and no B-frame.
- *Camera*: *iSight* webcam delivering 640x480 resolution at 30fps video stream.
- *HMD*: 3DVisor [26]
- *Computer*: *PowerMac* with *2x2 GHz PowerPC G5* processor, *1Go DDR SDRAM* Memory and *ATI Radeon 9600 Pro* graphics card.
- *Chemical plant* : A boiler with valves.

Test methodology used for this project is explained in details in Appendix A. Globally, 10 subjects, which are not video experts, evaluate a pair of interactions with the above-mentioned method (Forced Choice). Initially, user is presented with the reference interaction so that he assimilates it. Then each of the best encoding configuration is determined for each bit rate and the two scenarios. Test results are presented in the next section.

3.6 Test Results and perception model proposition

This section exposes the results of the static scenario A and the dynamic scenario B presented in the previous section. The first scenario is evaluated by 10 observers and the second by 13. Results are discussed according to the perception factors presented in section 3.3.1:

- *Fluidity*
- *Quality* as level of details
- *Latency*

For each bit rate two values are computed:

- The maximum user preferred encoding configuration. It is the pair which is the most often selected as best encoding configuration for a given bit rate. Connecting all these points in the adaptation space drives to the *Maximum Path of Preference*.
- The average of the preferred encoding configurations for a given bit rate. Connecting these averages in the adaptation space drives to the *Weighted Path of Preference*.

For example the first column (bit rate=100kbps) of Table 38 (Appendix B.2) shows that three observers prefer encoding configuration $(6.27_{fr}, 5.08_q)$, five $(5.11_{fr}, 6.23_q)$, four $(2.26_{fr}, 14.1_q)$ and one $(1.47_{fr}, 21.59_q)$. Thus $(5.11_{fr}, 6.23_q)$ is the maximum user preferred encoding configuration. The average of the preferred encoding configurations is computed as follows:

$$framerate = \frac{3 * 6.27 + 5 * 5.11 + 4 * 2.26 + 1.47}{13} = 4.22fps \quad (16)$$

$$quality = \frac{3 * 5.08 + 5 * 6.23 + 4 * 14.1 + 21.59}{13} = 9.56\% \quad (17)$$

Then this point is rounded to the closest possible encoding configuration for the given bit rate. Thus the weighted preferred encoding configuration is $(3.79_{fr}, 8.4_q)$.

3.6.1 Static Scenario Results

Table 37 shows the static scenario results. The ten first lines correspond to observer's preferred encoding configurations. The 11^{th} line gives the maximum user preferred encoding configurations and the 12^{th} the weighted preferred encoding configurations.
Figure 30 shows the Maximum and the Weighted Path of Preference.

Figure 30 shows that the quality is the dominant parameter. Indeed, the frame rate has no noticeable effect on the human perception for a static video sequence. The frame rate is only perceptible when motion appears in the video. As explained in section 2.4.4 high quality generates latency. But as for frame rate, latency doesn't influence the user quality perception when video is static. Thus, all subjects tend to maximize the quality parameter.

3.6.2 Dynamic Scenario Results

Table 38 shows the dynamic scenario results. The thirteen first lines correspond to the observer's preferred encoding configurations. The 14^{th} line gives the maximum user prefered encoding configurations and the 15^{th} the weighted prefered encoding configurations.
Figure 31 shows the Maximum and the Weighted Path of Preference.

It appears from the performed tests for dynamic interaction that latency is the most annoying factor. Hence, user systematically rejects encoding configurations which give high latency. It is the case for high quality and high frame rates as explained in Section 2.4.4.
Another observation is that frame rate has an annoying effect for values lower than 10fps approximately . For this range of values, jitter has a negative impact on the perception of the user interaction. Thus, subject privileges fluidity rather than quality. But that seems not to be the case for the first column of Table 38 (100kbps). Indeed some privileges quality and others the frame rate. This is due to the fact that the preceding test to evaluate was the static scenario. Some user continued to evaluate the quality only (as for the static test)[14]. But general tendency shows that frame rate is prefered. This phenomenon cannot be found with bit rates.
On the other hand, frame rates higher than 10fps have a weaker effect on perception. Hence, subject privileges quality than fluidity.

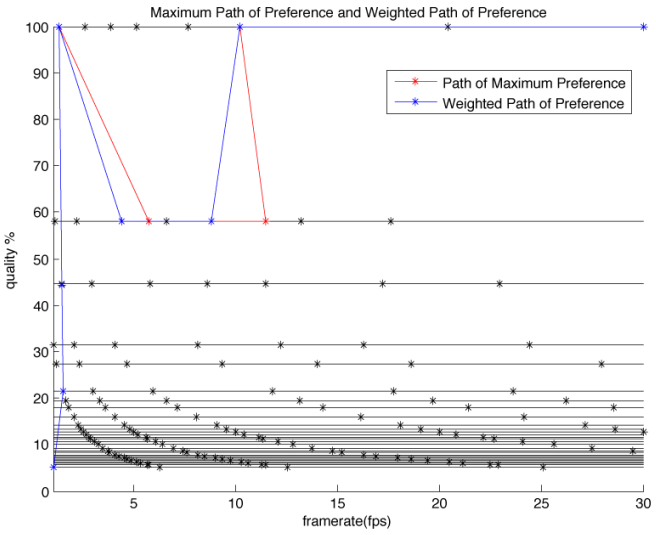

Fig. 30. Static Scenario: Maximum and Weighted Path of Preference

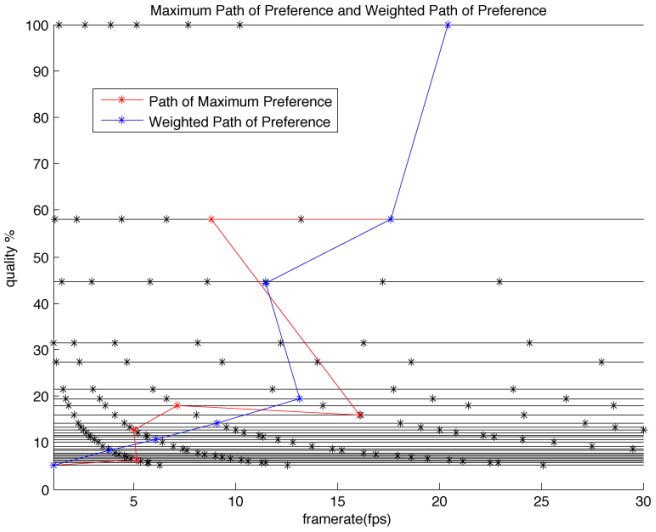

Fig. 31. Dynamic Scenario: Maximum and Weighted Path of Preference

14. To avoid this phenomenon scenario should be reversed

So, a global trend is derived from the performed tests. At first user rejects high qualities and frame rates to avoid latency. Then, for frame rates higher than 10fps, he privileges the quality and for lower values the frame rate is prefered. This can explain the special shape (*zigzag*) of the path. For example, the best encoding configurations at 800kbps and 1600kbps are respectively $((13.11_{fr}, 19.46_q))$ and $((11.48_{fr}, 44.44_q))$. At 800kbps, a decrease of the frame rate is necessary to increase the quality. But decreasing quality implies that video jitter is perceptible. So user prefers keeping a high frame rate rather than a high quality. On the other hand, at 1600kbps frame rate can be decreasing sufficiently without jitter appearance. Thus quality can be increased.

To refine the path between 800kbps and 3200kbps, two other bit rates, 1200kbps and 2400kbps are evaluated. The corresponding path is shown in Figure 32 which confirm the tendency described above.

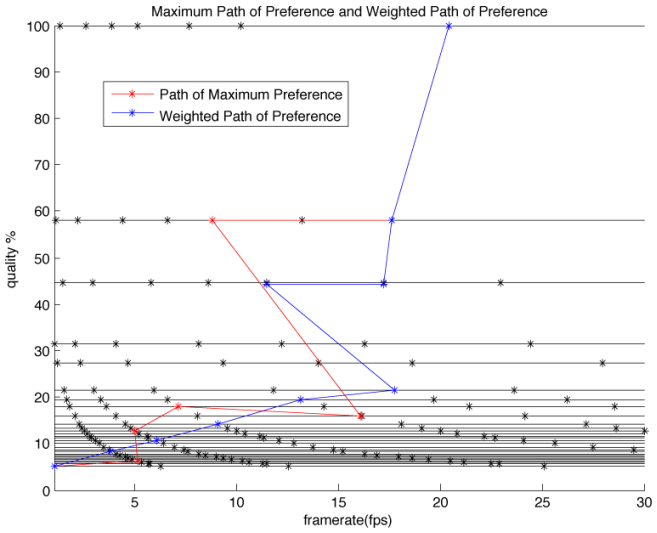

Fig. 32. Dynamic Scenario: Augmented Maximum and Weighted Path of Preference

3.6.3 LOAP and Perception Model Proposition

This section proposes a method to find the Local Optimal Adaptation Path from the weighted encoding configurations. Weighted points are prefered to Maximum points because they take all users evaluations into account. The method is based on the dynamic scenario results. Indeed, dynamic scenario reflects industrial reality better than the static scenario. Operators are generally active and it is rare that they are completely motionless. In addition, perception model based on dynamic scenario may be used for static operations. On the other hand the opposite is not valid because a model based on static scenario would maximize quality parameters without considering the frame rate. This would be harmful in term of quality of perception for active interactions.

A common mean to build a path through some points is to fit a continuous curve through them. But this solution is not adapted for this project because the adaptation space is not continuous (discrete quality values). Despite everything it is possible to find a continuous path in the adaptation space along frame rate axis. It can be done by moving along all bit rate values from the higher, 9400kbps given by $((30_{fr}, 100_q))$, to the the lowest, 16kbps given by $((1_{fr}, 5.08_q))$ and selecting an encoding configuration for each bit rate value. The method proposes to share path points between two successive preferred encoding configurations. If there are quality steps between these two values, then path points are also distributed on them. However, path points are distributed as less as possible on the highest quality because it provides annoying latency. The LOAP which corresponds to the perception model is shown in Figure 33. Reading this model from right to left and up to bottom corresponds to a continuous decrease of the bit rate.

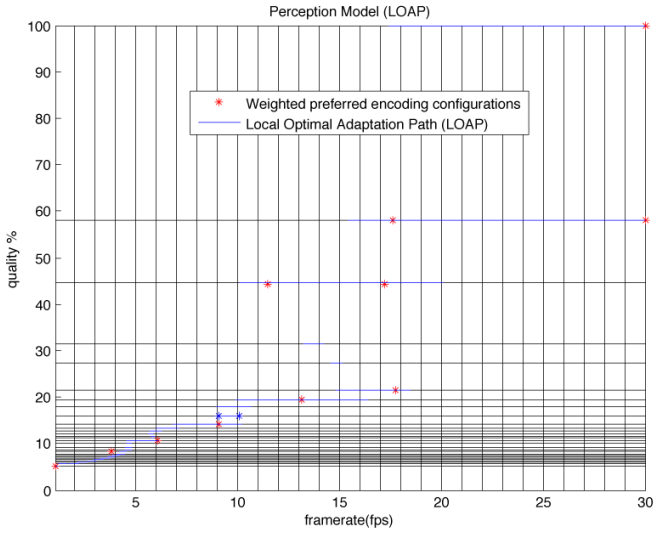

Fig. 33. LOAP: Perception Model

4 CONTROL

A Knowledge model is used in automatic to simulate the behavior of a physical system. If the model capture the reality fully and if the system is not subject to disturbances, then the control value to obtain a given system behavior is exactly known. In practice, knowledge model does not reflect exactly the system and disturbances are often present. Thus, a feedback of the output value and a comparison with the reference value are necessary to correct modeling errors and to reject disturbances.

In this project, the perception model does not simulate the behavior codec system itself but tells how to choose the best encoding parameters for a given bit rate to best match human visual system. However, the open-loop control system presented in Figure 34 is not satisfying because it is subjected to disturbances that perception model does not capture. Typical disturbance is a variation of the video content. For example, if the operator want to read a label, the scene becomes static. Thus, the bit rate provided by the codec strongly decreases. But open-loop control system is *blind* and cannot detect such a variation of the output bit rate. So the perception model keeps on supplying encoding parameters as if the scene was dynamic despite of the fact that it could provide parameters which would improve the user-perceived quality. Hence, to tackle this problem, a closed-loop control system is proposed in Figure 35 by adding a *Content Adaptation Scheme* (CAS).

The only way to act on the encoding parameters of the codec system is to change the input bit rate of the *Local Adaptation Scheme* (LAS). So the bit rate can be seen as the control value of a global system formed with the LAS, the perception model

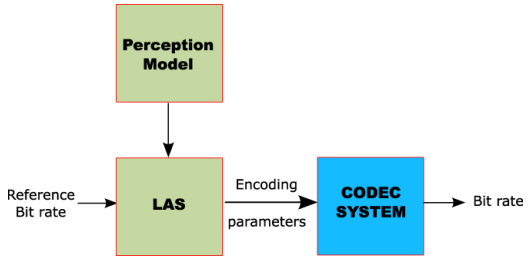

Fig. 34. Open Loop control system with perception model

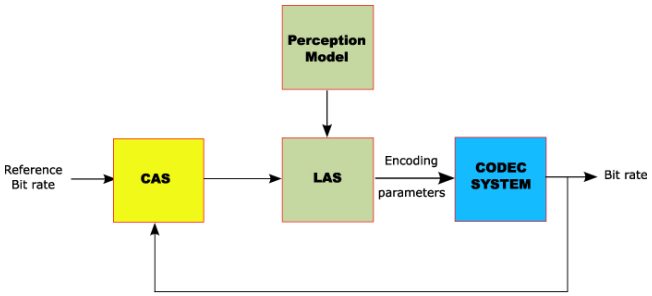

Fig. 35. Closed-Loop Control System

and the codec system. The perception model provides through the LAS the best encoding configurations for a given bit rate. Thus the reference bit rate work as an operating point and is directly provided to the global system. A PID controller, is then added to reject disturbances by acting on the bit rate error round the operating point. Figure 36 shows the general closed-loop control system and the content of the *Content Adaptation Scheme*.

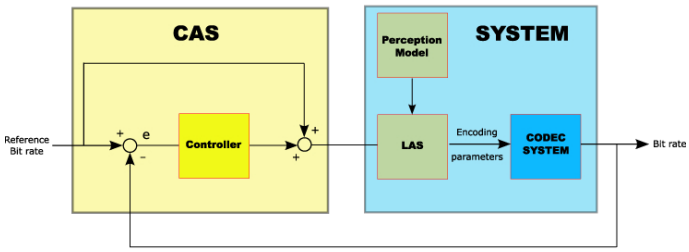

Fig. 36. Closed-Loop Control System

5 Conclusion

This project presents a general methodology to find a perception model in the context of chemical industry. This model is then used to implement a control schema that select the encoding parameters to provide the user with the best possible perception.
The perception model corresponds to an *Optimal Adaptation Path* (OAP) in the encoding parameters space. Subjective tests methodology were used to find the OAP instead of objective metrics. Subjective tests performed show that latency is the most annoying factor in human interaction. The perception model proposed is a model developed for the chemical plant context. More sophisticated models which take video content into account could be developped. Indeed, video content type strongly influence the bit rate delivering by the compression system. As proposed by Nicola Cranley [17], a solution to tackle this problem is to build a model for each content type. This drives to a content space in which an area corresponds to a different OAP in the adaptation space. But this solution needs to locate the video content in the content space in real time and this process is time-consuming and add latency.
A closed-loop control system is proposed to control the codec by taking advantage of the perception model. The concept of a controller is mentioned to reject video perturbations given by the video content itself and to compensated modeling error. The next step is to integrate the controller within the global adaptation scheme that include the network transmission. Thus, the global scheme takes into account both the QoS measured at the network level and the QoP measured at the human level.

Appendix A
Test Procedure

The test procedure is the following:

- **Introduction to user**
  Global explanation of the project and explanation of what is waited of users

  The goal of this project is to assess the quality of the video displayed on these glasses for different encoding configurations. Your mission is to interact with this boiler and to evaluate the global perception for a given operation.

- **Setup Presentation**
  This presentation permits the observer to familiarize with the interaction environment. Two types of interaction are evaluated (these comes from typicall industrial operations):
  1) Reading of a value (static test)
     Read the label 'BATIMENT 329'.
  2) Purging the boiler (dynamic test)
     Purge the boiler in the container. The procedure consists in opening the valve of decompression, the purging valve, filling the container with water, then closing the purging valve and finally closing the first opening valve.

  Then user may achieve some tests without the Head Mounted Display in order to familiarize with the requested operations.

- **Test Instructions**
  The instructions are the same for the two scenarios.

  At first you are requested to achieve the test with an optimal quality (reference sequence) in order to familiarize with the procedure. Then you will repeat twice the operation but with different compression configurations. Two consecutive tests are separated by a green screen. You are then requested to specify which sequence is better for you in term of global perception. You must take into account all factors: delay, jerk, quality. Answer as quickly as possible by taking in minds the two sequences only (and not the preceding). In the case where you cannot tell which was better, you must take a decision. Do you have any questions at this stage?

- **Equipment and calibration**
  The subject may then wear the Head Mounted Display (HMD) and adjust it so that the port is pleasant. In order to improve comfort the cables are lightened using a belt

around the bust. Finally video from camera is displayed at full quality and the user adjust the slop of the camera so that perceived video is aligned with the head.

- **Evaluation Procedure**

```
For(M=1; M<Number of scenarios; M++)
{
   Scenario M with optimal quality video
   for pre-evaluation (q=75.38% and fr=30fps);
   For(B=1; B < Number of selected bit rates; B++)
   {
     For(C=1; C<Number of couples to evaluate; C++)
     {
       Operation C.1;
       Green Screen;
       Operation C.2;
       What is the most pleasant operation?
       if(Are you tired?)
          Pauses or end of the test;
     }
   }
}
```

APPENDIX B
TEST RESULTS

B.1 Static Scenario A

Static Scenario A						
bitr. kbps	100	200	400	800	1600	3200
Obs.						
1	(1.47,21.59)	(2.95,21.59)	(1.27,100)	(2.55,100)	(5.1,100)	(17.6,58.02)
2	(1.47,21.59)	(1.43,44.44)	(1.27,100)	(2.55,100)	(11.48,44.44)	(10.21,100)
3	(1.47,21.59)	(1.43,44.44)	(1.27,100)	(5.74,44.44)	(11.48,44.44)	(17.6,58.02)
4	(1.01,31.38)	(1.43,44.44)	(1.27,100)	(5.74,44.44)	(5.1,100)	(17.6,58.02)
5	(1.47,21.59)	(1.43,44.44)	(1.27,100)	(5.74,44.44)	(11.48,44.44)	(17.6,58.02)
6	(1.47,21.59)	(1.43,44.44)	(1.27,100)	(2.55,100)	(11.48,44.44)	(17.6,58.02)
7	(1.47,21.59)	(1.43,44.44)	(1.27,100)	(2.55,100)	(11.48,44.44)	(10.21,100)
8	(1.47,21.59)	(1.43,44.44)	(1.27,100)	(5.74,44.44)	(11.48,44.44)	(10.21,100)
9	(1.47,21.59)	(1.43,44.44)	(1.27,100)	(5.74,44.44)	(8.8,58.02)	(10.21,100)
10	(1.47,21.59)	(1.43,44.44)	(1.27,100)	(2.55,100)	(5.1,100)	(10.21,100)
Max Path	(1.47,21.59)	(1.43,44.44)	(1.27,100)	(5.74,44.44)	(11.48,44.44)	(10.21,100)
W. Path	(1.47,21.59)	(1.43,44.44)	(1.27,100)	(4.4,58.02)	(8.8,58.02)	(10.21,100)

Fig. 37. Static Scenario A Results with all user prefered encoding configurations $(framerate[fps], quality[\%])$

B.2 Dynamic Scenario B

Dynamic Scenario B						
bitr. kbps	100	200	400	800	1600	3200
Obs.						
1	(6.27,5.08)	(8.13,7.85)	(11.09,11.51)	(16.1,15.85)	(16.27,31.38)	(17.6,58.02)
2	(2.26,14.1)	(8.13,7.85)	(7.13,17.89)	(16.1,15.85)	(23.64,21.59)	(17.6,58.02)
3	(2.26,14.1)	(8.13,7.85)	(7.13,17.89)	(16.1,15.85)	(11.48,44.44)	(17.6,58.02)
4	(5.11,6.23)	(8.13,7.85)	(7.13,17.89)	(11.82,21.59)	(8.8,58.02)	(17.6,58.02)
5	(2.26,14.1)	(2.95,21.59)	(11.09,11.51)	(16.1,15.85)	(11.48,44.44)	(10.21,100)
6	(5.11,6.23)	(8.13,7.85)	(7.13,17.89)	(11.82,21.59)	(8.8,58.02)	(17.6,58.02)
7	(6.27,5.08)	(5,12.76)	(7.13,17.89)	(8.13,31.38)	(8.8,58.02)	(10.21,100)
8	(5.11,6.23)	(5,12.76)	(7.13,17.89)	(11.82,21.59)	(8.8,58.02)	(22.97,44.44)
9	(2.26,14.1)	(5,12.76)	(7.13,17.89)	(8.13,31.38)	(8.8,58.02)	(22.97,44.44)
10	(1.47,21.59)	(5,12.76)	(11.09,11.51)	(11.82,21.59)	(8.8,58.02)	(22.97,44.44)
11	(5.11,6.23)	(5,12.76)	(7.13,17.89)	(11.82,21.59)	(16.27,31.38)	(17.6,58.02)
12	(6.27,5.08)	(8.13,7.85)	(11.09,11.51)	(16.1,15.85)	(16.27,31.38)	(17.6,58.02)
13	(5.11,6.23)	(5,12.76)	(7.13,17.89)	(16.1,15.85)	(23.64,21.59)	(17.6,58.02)
Max Path	(5.11,6.23)	(5,12.76)	(7.13,17.89)	(16.1,15.85)	(8.8,58.02)	(17.6,58.02)
W. Path	(3.79,8.4)	(6.02,10.59)	(9.04,14.01)	(13.11,19.46)	(11.48,44.44)	(17.6,58.02)

Fig. 38. Dynamic Scenario B Results with all user prefered encoding configurations $(framerate[fps], quality[\%])$

ACKNOWLEDGMENT

The author would like to thank Damien Perritaz, Christophe Salzmann and Philippe Cuanillon for their availability.

REFERENCES

[1] YUV, http://en.wikipedia.org/wiki/YUV
[2] Chroma Subsampling, http://en.wikipedia.org/wiki/Chroma_Subsampling
[3] Universal Serial Bus, http://en.wikipedia.org/wiki/Usb
[4] IEEE 802.11, http://en.wikipedia.org/wiki/802.11
[5] ITU Telecommunication Standardisation Sector, ITU-T, http://www.itu.int/ITU-T
[6] The MPEG Home Page, http://www.chiariglione.org/mpeg
[7] Video Codec, http://en.wikipedia.org/wiki/Video_codec
[8] Ghanbari M., 'Video coding: an introduction to standard codecs', *IEE Telecommunication series 42* (1999)
[9] JPEG Standard, JPEG ISO/IEC 10918-1 ITU-T Recommendation T.81, http://www.w3.org/Graphics/JPEG/itu-t81.pdf
[10] V. Bhaskaran, K. Konstantinides, 'Image and Video Compression Standards', Kluwer Academic Publishers, 1995.
[11] Feng Xiao, 'DCT-based Video Quality Evaluation', *Final Project for EE392J (2000)*, ise.stanford.edu/class/ee392j/projects/projects/xiao_report.pdf
[12] PortVideo, http://www.iua.upf.es/mtg/reacTable/?portvideo
[13] Daniel Heckenberg, 'seeSaw Code', http://www.cse.unsw.edu.au/ danielh/seeSaw/
[14] Daniel Heckenberg, 'Using Mac OS X for Real-Time Processing', http://www.cse.unsw.edu.au/ danielh/seeSaw/
[15] FFmpeg home page, http://ffmpeg.mplayerhq.hu/
[16] Cranley, N., Murphy, L., Perry, P., 'Dynamic Content-Based Adaptation of Streamed Multimedia', San Diego, CA, October, 2004.
[17] Cranley, N., 'User-Perceived Quality-Aware Adaptation of Streamed Multimedia over Best-effort IP Networks', National University of Ireland, March 2004.
[18] S.Winkler, 'Digital Video Quality: Vision Models And Metrics', John Wiley & Sons, 2005.
[19] Video Quality Experts Group (VQEG), http://www.its.bldrdoc.gov/vqeg/
[20] S. Wolf, M. Pinson, 'Spatial-temporal distortion metrics for in-service quality monitoring on any digital video system,' SPIE International Symposium on Voice, Video, and Data Communications, Boston, MA, September 11-22, 1999.
[21] A.P. Hekstra, J.G. Beerends, et al. 'PVQM - A perceptual video quality measure ', Signal Processing: Image Communication, Vol. 17, no. 10, 2002, pp. 781-798, 2002 Elsevier Science B.V
[22] Christian J. van den Branden Lambrecht and Olivier Verscheure, 'Perceptual Quality Measure using a Spatio-Temporal Model of the Human Visual System', Proceedings of SPIE 96, San Jose, CA
[23] ITU-R Recommendation BT.500-11, 'Methodology for the subjective assessment of the quality of television picture applications'
[24] M. Pinson and S. Wolf, 'Comparing subjective video quality testing methodologies' , SPIE Video Communications and Image Processing Conference, Lugano, Switzerland, Jul. 8-11 2003.
[25] ' Weber's Law of Just Noticeable Differences', http://en.wikipedia.org/wiki/Weber%27s_law
[26] 3DVisor Head Mounted Display, http://www.3dvisor.com/