CHAPTER 4

Video Quality Metrics

Quality generally indicates excellence, and the universal societal norm is to strive for the highest achievable quality in most fields. However, in case of digital video, a measured, careful approach is taken to allow some deficit in quality that is not always discernible by typical viewers. Such concessions in perceivable visual quality make room for a valuable accomplishment in terms of compression.

Video quality is a characteristic of a video signal passed through a transmission or processing system, representing a measure of perceived degradation with respect to the original source video. Video processing systems usually introduce some distortions or artifacts in the video signal, but the amount involved may differ depending on the complexity of the content and the parameters chosen to process it. The variable degradation may or may not be perceivable or acceptable to an end user. In general, it is difficult to determine what would be an acceptable quality for all end users. However, it remains an important objective of video quality evaluation studies. So understanding various types of visual degradations or artifacts in terms of their *annoyance factors*, and the evaluation of the quality of a video as apparent to the end user, are very important.

In this chapter, we first focus on the careful and intentional information loss due to compression and the resulting artifacts. Then we discuss the various factors involved in the compression and processing of video that influence the compression and that contribute to visual quality degradation.

With these understandings, we move toward measuring video quality and discuss various subjective and objective quality metrics to measure with particular attention to various ITU-T standards. The discussions include attempts to understand relative strengths and weaknesses of important metrics in terms of capturing perceptible deterioration. We further discuss video coding efficiency evaluation metrics and some example standard-based algorithms.

In the final part of this chapter, we discuss the parameters that primarily impact video quality, and which parameters need to be tuned to achieve good tradeoffs beween video quality and compression speed, the knowledge of which is useful in designing some video applications. Although some parameters are dictated by the available system or network resources, depending on the application, the end user may also be allowed to set or tune some of these parameters.

Compression Loss, Artifacts, and Visual Quality

Compression artifacts are noticeable distortions in compressed video, when it is subsequently decompressed and presented to a viewer. Such distortions can be present in compressed signals other than video as well. These distortions are caused by the lossy compression techniques involved. One of the goals of compression algorithms is to minimize the distortion while maximizing the amount of compression. However, depending on the algorithm and the amount of compression, the output has varying levels of diminishing quality or introduction of artifacts. Some quality-assessment algorithms can distinguish between distortions of little subjective importance and those objectionable to the viewer, and can take steps to optimize the final apparent visual quality.

Compression Loss: Quantization Noise

Compression loss is manifested in many different ways and results in some sort of visual impairment. In this section we discuss the most common form of compression loss and its related artifact, namely the quantization noise.

Quantization is the process of mapping a large set of input values to a smaller set—for example, rounding the input values to some unit of precision. A device or an algorithmic function that performs the quantization is called a quantizer. The round-off error introduced by the process is referred to as *quantization error* or the *quantization noise*. In other words, the difference between the input signal and the quantized signal is the quantization error.

There are two major sources of quantization noise in video applications: first, when an analog signal is converted to digital format; and second, when high-frequency components are discarded during a lossy compression of the digital signal. In the following discussion both of these are elaborated.

Quantization of Samples

The digitization process of an image converts the continuous-valued brightness information of each sample at the sensor to a discrete set of integers representing distinct gray levels—that is, the sampled image is quantized to these levels. The entire process of measuring and quantizing the brightnesses is significantly affected by sensor characteristics such as dynamic range and linearity. Real sensors have a limited dynamic range; they only respond to light intensity between some minimum and maximum values. Real sensors are also non-linear, but there may be some regions over which they are more or less linear, with non-linear regions at either end.

The number of various levels of quantizer output is determined by the bits available for quantization at the analog-to-digital converter. A quantizer with n bits represents $N = 2^n$ levels. Typical quantizers use 8-bits, representing 256 gray levels usually numbered between 0 and 255, where 0 corresponds to black and 255 corresponds to white. However, 10-bit or even 16-bit images are increasingly popular. Using more bits brings the ability to perform quantization with a finer step size, resulting in less noise and a closer approximation of the original signal. Figure 4-1 shows an example of a 2-bit or four-level quantized signal, which is a coarse approximation of the input signal, and a 3-bit or eight-level quantized signal, representing a closer approximation of the input signal.

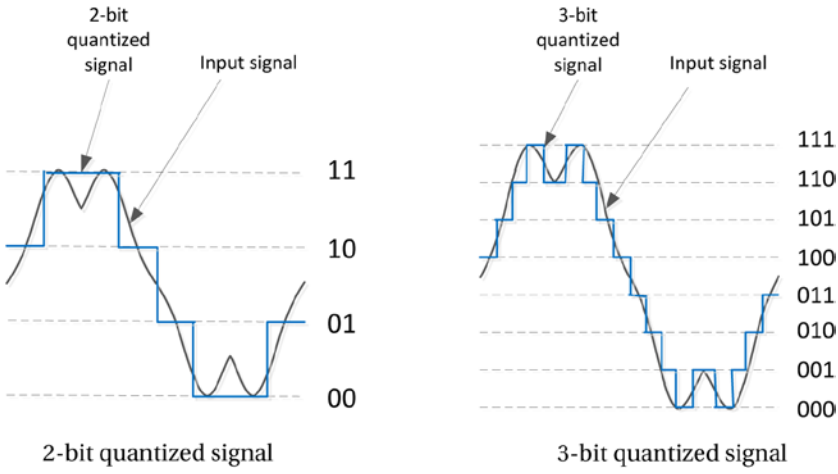


Figure 4-1. *Quantized signals with different bit resolution*

In the case of an image, the difference between the true input brightness of a pixel and the corresponding brightness of the digital level represents the quantization error for that pixel. Quantization errors can take positive or negative values. Note that quantization levels are equally spaced for uniform quantization, but are irregularly spaced for non-uniform (or non-linear) quantization. If the quantization levels are equally spaced with a step size b, the quantization error for a digital image may be approximated as a uniformly distributed signal with zero mean and a variance of $\frac{b^2}{12}$.

Such uniform quantizers are typically memoryless—that is, the quantization level for a pixel is computed independently of other pixels.

Frequency Quantization

In frequency quantization, an image or a video frame undergoes a transform, such as the discrete cosine transform, to convert the image into the frequency domain. For an 8×8 pixel block, 64 transform coefficients are produced. However, lossy compression techniques such as those adopted by the standards as described in Chapter 3, perform quantization on these transform coefficients using a same-size quantization matrix, which typically has non-linear scaling factors biased toward attenuating high-frequency components more than low-frequency components. In practice, most high-frequency components become zero after quantization. This helps compression, but the high-frequency components are lost irreversibly. During decompression, the quantized coefficients undergo inverse quantization operation, but the original values cannot be restored. The difference between the original pixel block and the reconstructed pixel block represents the amount of quantization error that was introduced. Figure 4-2 illustrates the concept.

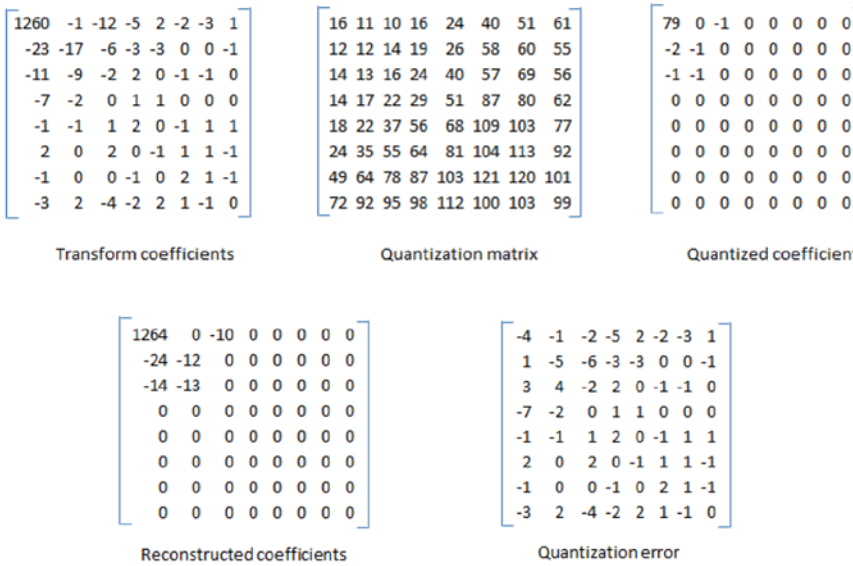


Figure 4-2. *Quantization of a block of transform coefficients*

The quantization matrix is the same size as the block of transform coefficients, which is input to the quantizer. To obtain quantized coefficients, an element-by-element division operation is performed, followed by a rounding to the nearest integer. For example, in Figure 4-2, quantization of the DC coefficient (the upper left element) by doing `round(1260/16)` gives the quantized coefficient 79. Notice that after quantization, mainly low-frequency coefficients, located toward the upper left-hand corner, are retained, while high-frequency coefficients have become zero and are discarded before transmission. Reconstruction is performed by multiplying the quantized coefficients by the same quantization matrix elements. However, the resultant reconstruction contains the quantization error as shown in Figure 4-2.

Usually quantization of a coefficient in a block depends on how its neighboring coefficients are quantized. In such cases, neighborhood context is usually saved and considered before quantizing the next coefficient. This is an example of a quantizer with memory.

It should be noted that the large number of zeros that appear in the quantized coefficients matrix is not by accident; the quantization matrix is designed in such a way that the high-frequency components–which are not very noticeable to the HVS–are removed from the signal. This allows greater compression of the video signal with little or no perceptual degradation in quality.

Color Quantization

Color quantization is a method to reduce the number of colors in an image. As the HVS is less sensitive to loss in color information, this is an efficient compression technique. Further, color quantization is useful for devices with limited color support. It is common to combine color quantization techniques, such as the nearest color algorithm, with

dithering–a technique for randomization of quantization error–to produce an impression of more colors than is actually available and to prevent color banding artifacts where continuous gradation of color tone is replaced by several regions of fewer tones with sudden tone changes.

Common Artifacts

Here are a few common artifacts that are typically found in various image and video compression applications.

Blurring Artifact

Blurring of an image refers to a smoothing of its details and edges, and it results from direct or indirect low-pass filter effects of various processing. Blurring of an object appears as though the object is out of focus. Generally speaking, blurring is an artifact the viewer would like to avoid, as clearer, crisper images are more desirable. But sometimes, blurring is intentionally introduced by using a Gaussian function to reduce image noise or to enhance image structures at different scales. Typically, this is done as a pre-processing step before compression algorithms may be applied, attenuating high-frequency signals and resulting in more efficient compression. This is also useful in edge-detection algorithms, which are sensitive to noisy environments. Figure 4-3 shows an example of blurring.

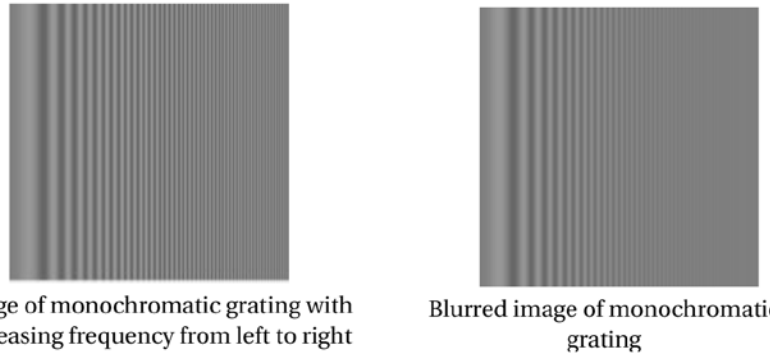


Figure 4-3. *An example of blurring of a frequency ramp. Low-frequency areas are barely affected by blurring, but the impact is visible in high-freqeuncy regions*

Motion blur appears in the direction of motion corresponding to rapidly moving objects in a still image or a video. It happens when the image being recorded changes position (or the camera moves) during the recording of a single frame, because of either rapid movement of objects or long exposure of slow-moving objects. For example, motion blur is often an artifact in sports content with fast motion. However, in sports contents,

motion blur is not always desirable; it can be inconvenient because it may obscure the exact position of a projectile or athlete in slow motion. One way to avoid motion blur is by panning the camera to track the moving objects, so the object remains sharp but the background is blurred instead. Graphics, image, or video editing tools may also generate the motion blur effect for artistic reasons; the most frequent synthetic motion blur is found when computer-generated imagery (CGI) is added to a scene in order to match existing real-world blur or to convey a sense of speed for the objects in motion. Figure 4-4 shows an example of motion blur.

Figure 4-4. *An example of motion blur*

Deinterlacing by the display and telecine processing by studios can soften images, and/or introduce motion-speed irregularities. Also, compression artifacts present in digital video streams can contribute additional blur during fast motion. Motion blur has been a more severe problem for LCD displays, owing to their sample-and-hold nature, where a continuous signal is sampled and the sample values are held for a certain time to eliminate input signal variations. In these displays, the impact of motion blur can be reduced by controlling the backlight.

Block Boundary Artifact

Block-based lossy coding schemes, including all major video and image coding standards, introduce visible artifacts at the boundaries of pixel blocks at low bit rates. In block-based transform coding, a pixel block is transformed to frequency domain using discrete cosine transform or similar transforms, and a quantization process discards the high-frequency coefficients. The lower the bit rate, the more coarsely the block is quantized, producing blurry, low-resolution versions of the block. In the extreme case, only the DC coefficient, representing the average of the data, is left for a block, so that the reconstructed block is only a single color region.

The *block boundary artifact* is the result of independently quantizing the blocks of transform coefficients. Neighboring blocks quantize the coefficients separately, leading to discontinuities in the reconstructed block boundaries. These block-boundary discontinuities are usually visible, especially in the flat color regions such as the sky, faces, and so on, where there are little details to mask the discontinuity. Compression algorithms usually perform deblocking operations to smooth out the reconstructed block boundaries, particularly to use a reference frame that is free from this artifact. Figure 4-5 shows an example of block boundary artifact.

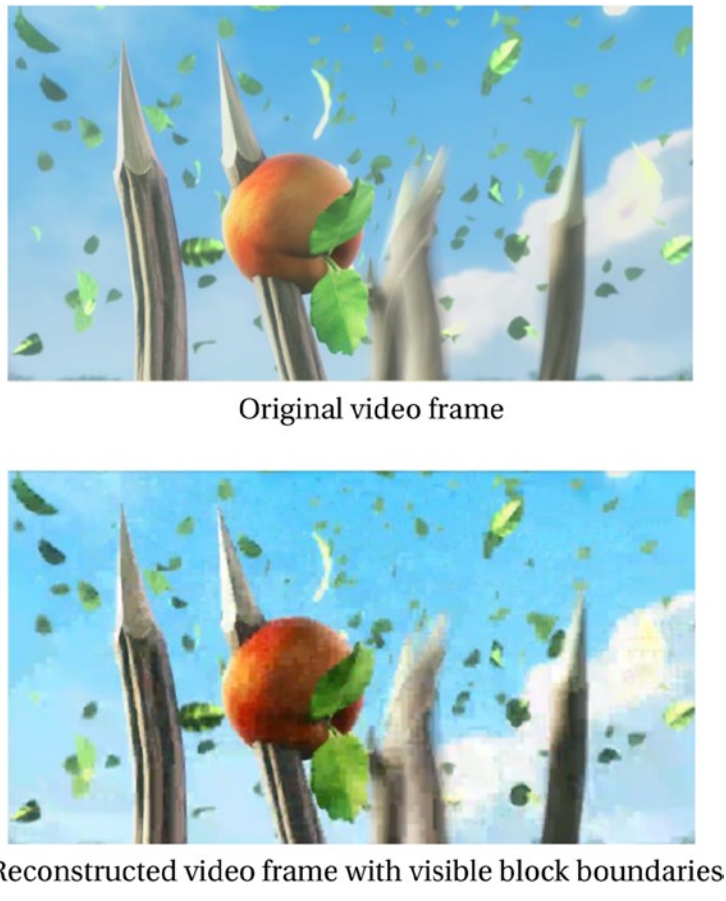
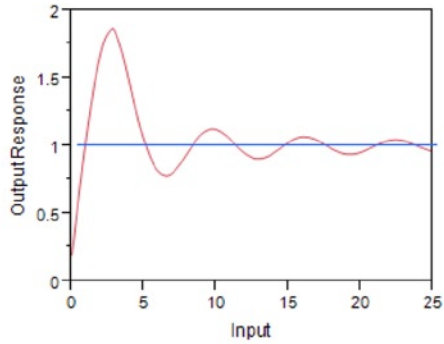


Figure 4-5. *An example of block boundary artifact*

This artifact is so common that many names are popularly used for it. Although the discontinuities may or may not align with the boundaries of macroblocks as defined in the video and image coding standards, *macroblocking* is a common term for this artifact. Other names include *tiling, mosaicing, quilting,* and *checkerboarding*.

Ringing Artifact

Ringing is unwanted oscillation of an output signal in response to a sudden change in the input. Image and video signals in digital data compression and processing are band limited. When they undergo frequency domain techniques such as Fourier or wavelet transforms, or non-monotone filters such as deconvolution, a spurious and visible *ghosting* or *echo* effect is produced near the sharp transitions or object contours. This is due to the well-known Gibb's phenomenon—an oscillating behavior of the filter's impulse response near discontinuities, in which the output takes higher value (*overshoots*) or lower value (*undershoots*) than the corresponding input values, with decreasing magnitude until a steady-state is reached. The output signal oscillates at a fading rate, similar to a bell ringing after being struck, inspiring the name of the *ringing artifact*. Figure 4-6 depicts the oscillating behavior of an example output response showing the Gibb's phenomenon. It also depicts an example of ringing artifact in an image.



Oscillating output in Gibb's phenomenon

Original image

Image with ringing artifact

Figure 4-6. *An example of ringing artifact*

Aliasing Artifacts

Let us consider a time-varying signal $x(t)$ and its sampled version $x(n) = x(nT)$, with sampling period $T>0$. When $x(n)$ is downsampled by a factor of 2, every other sample is discarded. In the frequency (ω) domain, the Fourier transform of the signal $X(e^{j\omega})$ is stretched by the same factor of 2. In doing so, the transformed signal can in general overlap with its shifted replicas. In case of such overlap, the original signal cannot be unambiguously recovered from its downsampled version, as the overlapped region represents two copies of the transformed signal at the same time. One of these copies is an *alias,* or replica of the other. This overlapping effect is called *aliasing*. Figure 4-7 shows the transform domain effect of downsampling, including aliasing.

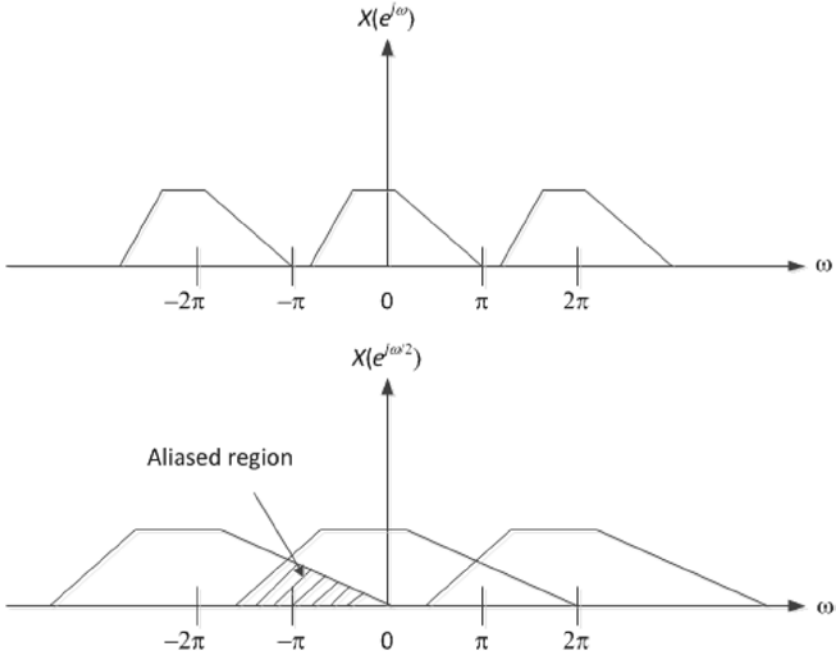


Figure 4-7. *Transform domain effect of downsampling, causing aliasing*

In general, aliasing refers to the artifact or distortion resulting from ambiguous reconstruction of signals from its samples. Aliasing can occur in signals sampled in time—for instance, digital audio—and is referred to as *temporal aliasing*. Aliasing can also occur in spatially sampled signals—for instance, digital images or videos—where it is referred to as *spatial aliasing*.

Aliasing always occurs when actual signals with finite duration are sampled. This is because the frequency content of these functions has no upper bound, causing their Fourier transform representation to always overlap with other transformed functions. On the other hand, functions with bounded frequency content (*bandlimited*) have infinite

duration. If sampled at a high rate above the so-called *Nyquist rate*, the original signal can be completely recovered from the samples. From Figure 4-7, it is clear that aliasing can be avoided if the original signal is bandlimited to the region $|\omega| < \frac{\pi}{M}$, where M is the downsampling factor. In this case, the original signal can be recovered from the downsampled version using an upsampler, followed by filtering.

Jaggies

Popularly known as *jaggies*, this common form of aliasing artifact produces visible stairlike lines where there should be smooth straight lines or curves in a digital image. These stairs or steps are a consequence of the regular, square layout of a pixel. With increasing image resolution, this artifact becomes less visible. Also, anti-aliasing filters are useful in reducing the visibility of the aliased edges, while sharpening increases such visibility.

Figure 4-8 shows examples of aliasing artifacts such as jaggies and moiré patterns.

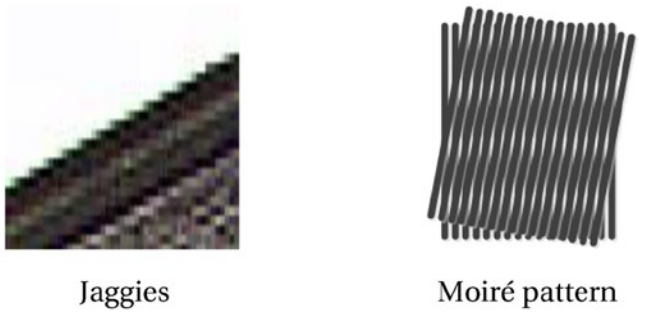


Figure 4-8. *Examples of aliasing artifacts*

Moiré Pattern

Due to undersampling of a fine regular pattern, a special case of aliasing occurs in the form of *moiré patterns*. It is an undesired artifact of images produced by various digital imaging and computer graphics techniques—for example, ray tracing a checkered surface. The moiré effect is the visual perception of a distinctly different third pattern, which is caused by inexact superposition of two similar patterns. In Figure 4-8, moiré effect can be seen as an undulating pattern, while the original pattern comprises a closely spaced grid of straight lines.

Flickering Artifacts

Flicker is perceivable interruption in brightness for a sufficiently long time (e.g., around 100 milliseconds) during display of a video. It is a flashing effect that is displeasing to the eye. Flicker occurs on old displays such as cathode ray tubes (CRT) when they are driven at a low refresh rate. Since the shutters used in liquid crystal displays (LCD) for each pixel stay at a steady opacity, they do not flicker, even when the image is refreshed.

Jerkiness

A flicker-like artifact, *jerkiness* (also known as *choppiness*), describes the perception of individual still images in a motion picture. It may be noted that the frequency at which flicker and jerkiness are perceived is dependent upon many conditions, including ambient lighting conditions. Jerkiness is not discernible for normal playback of video at typical frame rates of 24 frames per second or above. However, in visual communication systems, if a video frame is dropped by the decoder owing to its late arrival, or if the decoding is unsuccessful owing to network errors, the previous frame would continue to be displayed. Upon successful decoding of the next error-free frame, the scene on the display would suddenly be updated. This would cause a visible jerkiness artifact.

Telecine Judder

Another flicker-like artifact is the *telecine judder*. In order to convert the 24 fps film content to 30 fps video, a process called telecine, or 2:3 pulldown, is commonly applied. The process converts every four frames of films to five frames of interlaced video. Some DVD or Blu-ray players, line doublers, or video recorders can detect telecine and apply a reverse telecine process to reconstruct the original 24 fps video content. Figure 4-9 shows the telecine process.

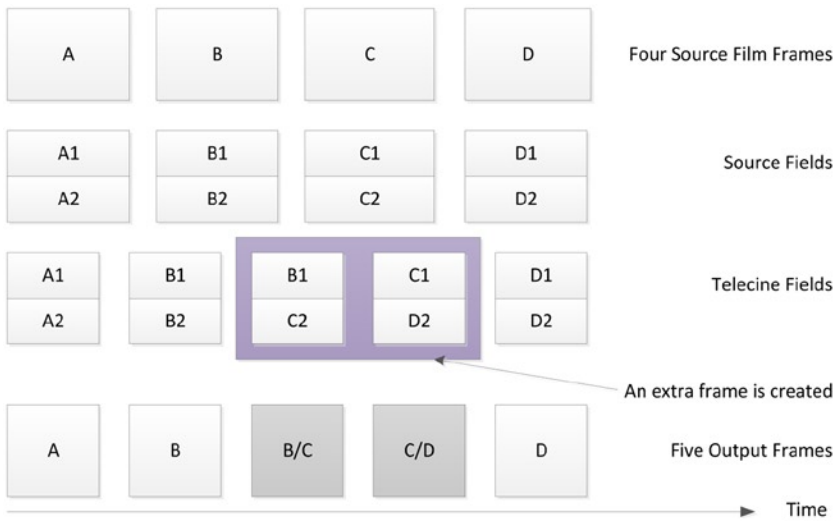


Figure 4-9. *The telecine process*

Notice that by the telecine process two new frames B/C and C/D are created, that were not part of the original set of source frames. Thus, the telecine process creates a slight error in the video signal compared to the original film frames. This used to create

the problem for films viewed on NTSC television that they would not appear as smooth as when viewed in a cinema. This problem was particularly visible during slow, steady camera movements that would appear slightly jerky when telecined.

Other Image Artifacts

There are several other artifacts commonly observed in compressed video. Some of these are discussed below. Figure 4-10 shows examples of various image artifacts.

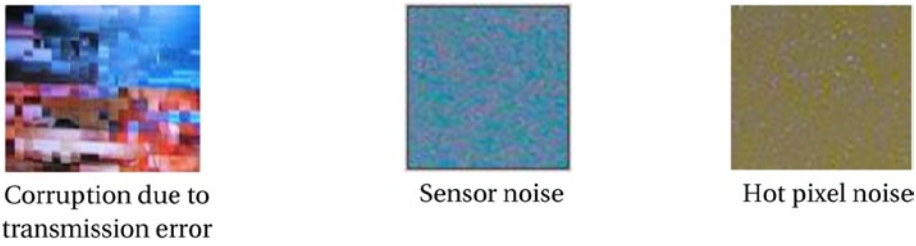


Figure 4-10. *Examples of various image artifacts*

Corruption due to Transmission Error

Owing to transmission errors in the compressed bitstream ,visible corruption may be observed in the reconstructed signal. Transmission errors can also disrupt the bitstream parsing, leading to partially decoded pictures or decoded pictures with missing blocks. In case of gross errors, decoders may continue to apply updates to the damaged picture for a short time, creating a *ghost image* effect, until the next error-free independently compressed frame is available. Ghosting is a common artifact in open-air television signals.

Image Noise

The camera sensor for each pixel contains one or more light-sensitive photodiodes that convert the incoming light into electrical signals, which is processed into the color value of the pixel in an image. However, this process is not always perfectly repeatable, and there are some statistical variations. Besides, even without incident light, the electrical activity of the sensors may generate some signal. These unwanted signals and variations are the sources of *image noise*. Such noise varies per pixel and over time, and increases with the temperature. Image noise can also originate from film grain.

Noise in digital images is most visible in uniform surfaces, such as in skies and shadows as monochromatic grain, and/or as colored waves (color noise). Another type of noise, commonly called *hot pixel* noise, occurs with long exposures lasting more than a second and appears as colored dots slightly larger than a single pixel. In modern cameras, however, hot pixel noise is increasingly rare.

Factors Affecting Visual Quality

Visual artifacts resulting from loss of information due to processing of digital video signals usually degrade the perceived visual quality. In addition to the visual artifacts described above, the following are important contributing factors that affect visual quality.

- **Sensor noise and pre-filtering:** Sensor noise, as mentioned above, is an undesirable by-product of image capture that affects visual quality. Not only is the noise itself visually disturbing, but its presence also impacts subsequent processing, causing or aggravating further artifacts. For example, pre-filtering is typically done after an image is captured but before encoding. In the pre-filtering stage, aliasing or ringing artifacts can occur; these artifacts would be visible even if lossless encoding is performed.

- **Characteristics of video:** Visual quality is affected by digital video characteristics including bit depth, resolution, frame rate, and frame complexity. Typical video frames use 8 bits for each pixel component, while premium quality videos allocate 10 to 16 bits. Similarly, high-definition video frames are four to six times as large as standard-definition video frames, depending on the format. Ultra-high-definition videos exhibit the highest quality owing to their 24 to 27 times higher pixel resolutions than their standard-definition counterparts.

  Frame rate is another important factor; although the HVS can perceive slow motion at 10 frames per second (fps) and smooth motion at 24 fps, higher frame rates imply smoother motion, especially for fast-moving objects. For example, a moving ball may be blurry at 30 fps, but would be clearer at 120 fps. Very fast motion is more demanding–wing movements of a hummingbird would be blurry at 30 fps, or even at 120 fps; for clear view of such fast motion, 1000 fps may be necessary. Higher frame rates are also used to produce special slow-motion effects. One measure of the complexity of a frame is the amount of details or *spatial business* of the frame. Artifacts in frames with low complexity and low details are generally more noticeable than frames with higher complexity.

  The spatial information (detail) and temporal information (motion) of the video are critical parameters. These play a crucial role in determining the amount of video compression that is possible and, consequently, the level of impairment that is suffered when the scene is transmitted over a fixed-rate digital transmission service channel.

- **Amount of compression:** For compressed digital video the amount of compression matters because compression is usually achieved by trading off visual quality. Highly compressed video has lower visual quality than lightly compressed video. Compression artifacts are noticeable and can be annoying for low-bit-rate video. Also, based on available bits, different amounts of quantization may have been done per block and per frame. The impact of such differences can be visible depending on the frame complexity. Furthermore, although compression techniques such as chroma subsampling take advantage of the HVS characteristics, premium contents with 4:4:4 chroma format have better visual quality compared to 4:2:0 contents.
- **Methods of compression:** Lossless compression retains all the information present in the video signal, so it does not introduce quality degradation. On the other hand, lossy compression aims to control the loss of quality by performing a careful tradeoff between visual quality and amount of compression. In lossy compression, selection of modes also influences the quality. In error-prone environments such as wireless networks, intra modes serve as a recovery point from errors at the expense of using more bits.
- **Multiple passes of processing:** In off-line video applications where real-time processing is not necessary, the video signal may undergo multiple passes. Analyzing the statistics of the first pass, parameters can be tuned for subsequent passes. Such techniques usually produce higher quality in the final resulting signals. However, artifacts due to various processing may still contribute to some quality loss.
- **Multiple generations of compression:** Some video applications may employ multiple generations of compression, where a compressed video signal is decompressed before compressing again with possibly different parameters. This may result in quality degradation owing to the use of different quantization maps for each generation. Typically, after the second generation visual quality deteriorates dramatically. To avoid such quality loss, robust design of quantization parameters is necessary.
- **Post-production:** Post-production effects and scene cuts can cause different portions of the encoded video sequence to have different quality levels.

Video Quality Evaluation Methods and Metrics

Video quality is evaluated for specification of system requirements, comparison of competing service offerings, transmission planning, network maintenance, client-based quality measurement, and so on. Several methods have been proposed in the literature to address the quality evaluation problem for various usages. With many methods and

algorithms available, the industry's need for accurate and reliable objective video metrics has generally been addressed by the ITU in several recommendations, each aiming toward particular industries such as standard- and high-definition broadcast TV.

The standardization efforts are being extended with the progress of modern usages like mobile broadcasting, Internet streaming video, IPTV, and the like. Standards address a variety of issues, including definitions and terms of reference, requirements, recommended practices, and test plans. In this section, we focus on the definitions, methods, and metrics for quality-evaluation algoithms. In particular, Quality of Experience (QoE) of video is addressed from the point of view of overall user experience—that is, the viewer's perception—as opposed to the well-known Quality of Service (QoS) measure usually employed in data transmission and network performance evaluation.

There are two approaches to interpreting video quality:

- The first approach is straightforward; the actual visual quality of the image or video content is determined based on *subjective* evaluation done by humans.
- In the second approach is synonymous with the signal fidelity or similarity with respect to a *reference* or *perfect* image in some perceptual space. There are sophisticated models to capture the statistics of the natural video signals; based on these models, *objective* signal fidelity criteria are developed that relate video quality with the amount of information shared between a reference and a distorted video signal.

In the following discussion, both subjective and objective video quality metrics are presented in detail.

Subjective Video Quality Evaluation

Video processing systems perform various tasks, including video signal acquisition, compression, restoration, enhancement, and reproduction. In each of these tasks, aiming for the best video quality under the constraints of the available system resources, the system designers typically make various tradeoffs based on some quality criteria.

An obvious way of measuring quality is to solicit the opinion of human observers or *subjects*. Therefore, the subjective evaluation method of video quality utilizes human subjects to perform the task of assessing visual quality. However, it is impossible to subjectively assess the quality of every video that an application may deal with. Besides, owing to inherent variability in quality judgment among human observers, multiple subjects are usually required for meaningful subjective studies. Furthermore, video quality is affected by viewing conditions such as ambient illumination, display device, and viewing distance. Therefore, subjective studies must be conducted in a carefully controlled environment.

Although the real perceptual video quality can be tracked by using this technique, the process is cumbersome, not automatable, and the results may vary depending on the viewer, as the same visual object is perceived differently by different individuals. Nevertheless, it remains a valuable method in providing ground-truth data that can be used as a reference for the evaluation of automatic or objective video quality-evaluation algorithms.

Objective algorithms estimate a viewer's perception, and the performance of an algorithm is evaluated against subjective test results. Media degradations impact the viewers' perception of the quality. Consequently, it is necessary to design subjective tests that can accurately capture the impact of these degradations on a viewer's perception. These subjective tests require performing comprehensive experiments that produce consistent results. The following aspects of subjective testing are required for accurate evaluation of an objective quality algorithm:

- Viewers should be naïve and non-expert, representing normal users whose perception is estimated by the objective quality models. These viewers vote on the subjective quality as instructed by the test designer. However, for specific applications, such as new codec developments, experienced voters are more suitable.
- The number of voters per sample should meet the subjective testing requirements as described in the appropriate ITU-T Recommendations. Typically a minimum of 24 voters is recommended.
- To maintain consistency and repeatability of experiments, and to align the quality range and distortion types, it is recommended that the experiments contain an anchor pool of samples that best represent the particular application under evaluation. However, it should be noted that even when anchor samples are used, a bias toward different experiments is common, simply because it is not always possible to include all distortion types in the anchor conditions.

Study group 9 (SG9) of ITU-T developed several recommendations, of which the Recommendation BT. 500-13[1] and the P-series recommendations are devoted to subjective and objective quality-assessment methods. These recommendations suggest standard viewing conditions, criteria for the selection of observers and test material, assessment procedures, and data analysis methods. Recommendations P.910[2] through P.913[3] deal with subjective video quality assessment for multimedia applications. Early versions, such as Rec. P.910 and P.911,[4] were designed around the paradigm of a fixed video service for multimedia applications. This paradigm considers video transmission over a reliable link to an immobile *cathode ray tube* (CRT) television located in a quiet and nondistracting environment, such as a living room or office. To accommodate new applications, such as Internet video and distribution quality video, P.913 was introduced.

---

[1] *ITU-R Recommendation BT.500-13: Methodology for the Subjective Assessment of the Quality of Television Pictures* (Geneva, Switzerland: International Telecommunications Union, 2012).
[2] *ITU-T Recommendation P.910: Subjective Video Quality Assessment Methods for Multimedia Applications* (Geneva, Switzerland: International Telecommunications Union, 2008).
[3] *ITU-T Recommendation P.913: Methods for the Subjective Assessment of Video Quality, Audio Quality and Audiovisual Quality of Internet Video and Distribution Quality Television in Any Environment* (Geneva, Switzerland: International Telecommunications Union, 2014).
[4] *ITU-T Recommendation P.911: Subjective Audiovisual Quality Assessment Methods for Multimedia Applications* (Geneva, Switzerland: International Telecommunications Union, 1998).

Ratified in January 2014, Recommendation P.913 describes non-interactive subjective assessment methods for evaluating the one-way overall video quality, audio quality, and/or audio-visual quality. It aims to cover a new paradigm of video—for example, an on-demand video service, transmitted over an unreliable link to a variety of mobile and immobile devices located in a distracting environment, using LCDs and other flat-screen displays. This new paradigm impacts key characteristics of the subjective test, such as the viewing environment, the listening environment, and the questions to be answered. Subjective quality assessment in the new paradigm asks questions that are not considered in the previous recommendations. However, this recommendation does not address the specialized needs of broadcasters and contribution quality television.

The duration, the number and type of test scenes, and the number of subjects are critical for the interpretation of the results of the subjective assessment. P.913 recommends stimuli ranging from 5 seconds to 20 seconds in duration, while 8- to 110-second sequences are highly recommended. Four to six scenes are considered sufficient when the variety of content is respected. P.913 mandates that at least 24 subjects must rate each stimulus in a controlled environment, while at least 35 subjects must be used in a public environment. Fewer subjects may be used for pilot studies to indicate trending.

Subjective Quality Evaluation Methods and Metrics

The ITU-T P-series recommendations define some of the most commonly used methods for subjective quality assessment. Some examples are presented in this section.

Absolute Category Rating

In the *absolute category rating* (ACR) method, the quality judgment is classified into several categories. The test stimuli are presented one at a time and are rated independently on a category scale. ACR is a single-stimulus method, where a viewer watches one stimulus (e.g., video clip) and then rates it. ACR methods are influenced by the subject's opinion of the content—for example, if the subject does not like the production of the content, he may give it a poor rating. The ACR method uses the following five-level rating scale:

5	Excellent
4	Good
3	Fair
2	Poor
1	Bad

A variant of the ACR method is *ACR with hidden reference* (ACR-HR). With ACR-HR, the experiment includes a reference version of each video segment, not as part of a pair but as a freestanding stimulus for rating. During the data analysis the ACR scores are subtracted from the corresponding reference scores to obtain a *differential viewer*

(DV) score. This procedure is known as *hidden reference removal*. The ACR-HR method removes some of the influence of content from the ACR ratings, but to a lesser extent than double-stimulus methods, which are discussed below.

Degradation Category Rating

Also known as the double-stimulus impair scale (DSIS) method, the *degradation category rating* (DCR) presents a pair of stimuli together. The reference stimulus is presented first, followed by a version after it has undergone processing and quality degradation. In this case, the subjects are asked to rate the impairment of the second stimulus with respect to the reference. DCR is minimally influenced by the subject's opinion of the content. Thus, DCR is able to detect color impairments and skipping errors that the ACR method may miss. However, DCR may have a slight bias, as the reference is always shown first.
In DCR, the following five-level scale for rating the relative impairment is used:

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

Comparison Category Rating

The *comparison category rating* (CCR) is a double-stimulus method whereby two versions of the same stimulus are presented in a randomized order. For example, half of the time the reference is shown first, and half the time it is shown second, but in random order. CCR is also known as the double-stimulus comparison scale (DSCS) method.
It may be used to compare reference video with processed video, or to compare two different impairments. CCR, like DCR, is minimally influenced by the subject's opinion of the content. However, occasionally subjects may inadvertently swap their rating in CCR, which would lead to a type of error that is not present in DCR or ACR. In CCR, the following seven-level scale is used for rating.

-3	Much worse
-2	Worse
-1	Slightly worse
0	The same
1	Slightly better
2	Better
3	Much better

SAMVIQ

Subjective assessment of multimedia video quality (SAMVIQ) is a non-interactive subjective assessment method used for video-only or audio-visual quality evaluation, spanning a large number of resolutions from SQCIF to HDTV. The SAMVIQ methodology uses a continuous quality scale. Each subject moves a slider on a continuous scale graded from zero to 100. This continuous scale is annotated by five quality items linearly arranged: excellent, good, fair, poor, and bad.

MOS

The *mean opinion score* (MOS) is the most common metric used in subjective video quality evaluation. It forms the basis of the subjective quality-evaluation methods, and it serves as a reference for the objective metrics as well. Historically, this metric has been used for decades in telephony networks to obtain the human user's view of the quality of the network. It has also been used as a subjective audio-quality measure. After all the subjects are run through an experiment, the ratings for each clip are averaged to compute either a MOS or a *differential mean opinion score* (DMOS).

The MOS provides a numerical indication of the perceived quality from the user's point of view of the received media after it has undergone compression and/or transmission. The MOS is generally expressed as a single number in the range from 1 to 5, where 1 is the lowest perceived quality, and 5 is the highest perceived quality. MOS is used for single-stimulus methods such as ACR or ACR-HR (using raw ACR scores), where the subject rates a stimulus in isolation. In contrast, the DMOS scores measure a change in quality between two versions of the same stimulus (e.g., the source video and a processed version of the video). ACR-HR (in case of average differential viewer score), DCR, and CCR methods usually produce DMOS scores.

Comparing the MOS values of different experiments requires careful consideration of intra- and inter-experimental variations. Normally, only the MOS values from the same test type can be compared. For instance, the MOS values from a subjective test that use an ACR scale cannot be directly compared to the MOS values from a DCR experiment. Further, even when MOS values from the same test types are compared, the fact that each experiment is slightly different even for the same participants leads to the following limitations:

- A score assigned by subject is rarely always the same, even when an experiment is repeated with the same samples in the same representation order. Usually this is considered as a type of noise on the MOS scores.
- There is a short-term context dependency as subjects are influenced by the short-term history of the samples they have previously scored. For example, following one or two poor samples, subjects tend to score a mediocre sample higher. If a mediocre sample follows very good samples, there is a tendency to score the mediocre sample lower. To average out this dependency, the presentation order should be varied for the individual subjects. However, this strategy does not remove the statistical uncertainty.

- The mid-term contexts associated with the average quality, the quality distribution, and the occurance of distortions significantly contribute to the variations between subjective experiments. For example, if an experiment is composed of primarily low-quality samples, people tend to score them higher, and vice versa. This is because people tend to use the full quality scale offered in an experiment and adapt the scale to the qualities presented in the experiment. Furthermore, individual distortions for less frequent samples are scored lower compared to experiments where samples are presented more often and people become more familiar with them.
- The long-term dependencies reflect the subject's cultural interpretation of the category labels, the cultural attitude to quality, and language dependencies. For example, some people may have more frequent experiences with video contents than others. Also, the expectations regarding quality may change over time. As people become familiar with digital video artifacts, it becomes part of their daily experience.

Although these effects cause differences between individual experiments, they cannot be avoided. However, their impacts can be minimized by providing informative instructions, well-balanced test designs, a sufficient number of participants, and a mixed presentation order.

Objective Video Quality Evaluation Methods and Metrics

Video quality assessment (VQA) studies aim to design algorithms that can automatically evaluate the quality of videos in a manner perceptually consistent with the subjective human evaluation. This approach tracks an *objective* video-quality metric, which is automatable, and the results are verifiable by repeated execution, as they do not require human field trial. However, these algorithms merely attempt to predict human subjective experience and are not perfect; they will fail for certain unpredictable content. Thus, the objective quality evaluation cannot replace subjective quality evaluation; they only aid as a tool in the quality assessment. The ITU-T P.1401[5] presents a framework for the statistical evaluation of objective quality algorithms regardless of the assessed media type.

In P.1401, the recommended statistical metrics for objective quality assessment need to cover three main aspects—accuracy, consistency, and linearity—against subjective data. It is recommended that the prediction error be used for accuracy, the outlier ratio or

[5]*ITU-T Recommendation P.1401: Methods, Metrics and Procedures for Statistical Evaluation, Qualification and Composition of Objective Quality Prediction Models* (Geneva, Switzerland: International Telecommunications Union, 2012).

the residual error distribution for consistency, and the Pearson correlation coefficient for linearity. The root mean square of the prediction error is given by:

$$RMSE\ of\ P_{error} = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(MOS(i) - MOS_{predicted}(i))^2} \qquad \text{(Equation 4-1)}$$

where N is the number of samples and N-1 ensures an unbiased estimator for the RMSE.

The distribution of the residual error ($MOS–MOS_{predicted}$) is usually characterized by a binomial distribution. The probability of exhibiting residual error below a pre-established threshold (usually 95% confidence interval) has a mean $P_{th} = (N_{th}/N)$, where N_{th} is number of samples for which residual error remains below the threshold, and a standard deviation

$$\sigma_{th} = \sqrt{\frac{P_{th}(1-P_{th})}{N}}\ .$$

An objective video or image quality metric can play a variety of roles in video applications. Notable among these are the following:

- An objective metric can be used to dynamically monitor and adjust the quality. For example, a network digital video server can appropriately allocate, control, and trade off the streaming resources based on the video quality assessment on the fly.
- It can be used to optimize algorithms and parameter settings of video processing systems. For instance, in a video encoder, a quality metric can facilitate the optimal design of pre-filtering and bit-rate control algorithms. In a video decoder, it can help optimize the reconstruction, error concealment, and post-filtering algorithms.
- It can be used to benchmark video processing systems and algorithms.
- It can be used to compare two video systems solutions.

Classification of Objective Video Quality Metrics

One way to classify the objective video quality evaluation methods is to put them into three categories based on the amount of reference information they require: *full reference* (FR), *reduced reference* (RR), and *no reference* (NR). These methods are discussed below. The FR methods can be further categorized as follows:

- Error sensitivity based approaches
- Structural similarity based approaches
- Information fidelity based aproaches
- Spatio-temporal approaches

- Saliency based approaches
- Network aware approaches

These approaches are discussed in the following subsections. Further, an example metric for each approach is elaborated.

Full Reference

A digital video signal undergoes several processing steps during which video quality may have been traded off in favor of compression, speed, or other criteria, resulting in a distorted signal that is available to the viewer. In objective quality assessment, the fidelity of the distorted signal is typically measured. To determine exactly how much degradation has occurred, such measurements are made with respect to a reference signal that is assumed to have *perfect* quality. However, the reference signal may not be always available.

Full-reference[6] (FR) metrics measure the visual quality degradation in a distorted video with respect to a reference video. They require the entire reference video to be available, usually in unimpaired and uncompressed form, and generally impose precise spatial and temporal alignment, as well as calibration of luminance and color between the two videos. This allows every pixel in every frame of one video to be directly compared with its counterpart in the other video.

Typically, the fidelity is determined by measuring the *distance* between the reference and the distorted signals in a perceptually meaningful way. The FR quality evaluation methods attempt to achieve consistency in quality prediction by modeling the significant physiological and psychovisual features of the HVS and using this model to evaluate signal fidelity. As fidelity increases, perceived quality of the content also increases. Although FR metrics are very effective in analysis of video quality, and are very widely used for analysis and benchmarking, the FR metrics' requirement that the reference be accessible during quality evaluation at the reconstruction end may not be fulfilled in practice. Thus, their usefulness may become limited in such cases.

Reduced Reference

It is possible to design models and evaluation criteria when a reference signal is not fully available. Research efforts in this area generated the various *reduced-reference*[7] (RR) methods that use partial reference information. They extract a number of features from the reference and/or the distorted test video. These features form the basis of the comparison between the two videos, so that the full reference is not necessary. This approach thus avoids the assumptions that must be made in the absence of any reference information, while keeping the amount of reference information manageable.

---

[6] *ITU-T Recommendation J.247: Objective Perceptual Multimedia Video Quality Measurement in the Presence of a Full Reference* (Geneva, Switzerland: International Telecommunications Union, 2009).

[7] *ITU-T Recommendation J.246: Perceptual Visual Quality Measurement Techniques for Multimedia Services over Digital Cable Television Networks in the Presence of a Reduced Bandwidth Reference* (Geneva, Switzerland: International Telecommunications Union, 2008).

No Reference

No-reference (NR) metrics analyze only the distorted test video without depending on an explicit reference video. As a result, NR metrics are not susceptible to alignment issues. The main challenge in NR approaches, however, is to distinguish between the distortions and the actual video signal. Therefore, NR metrics have to make assumptions about the video content and the types of distortions.

Figure 4-11 shows typical block diagrams of the FR and the NR approaches.

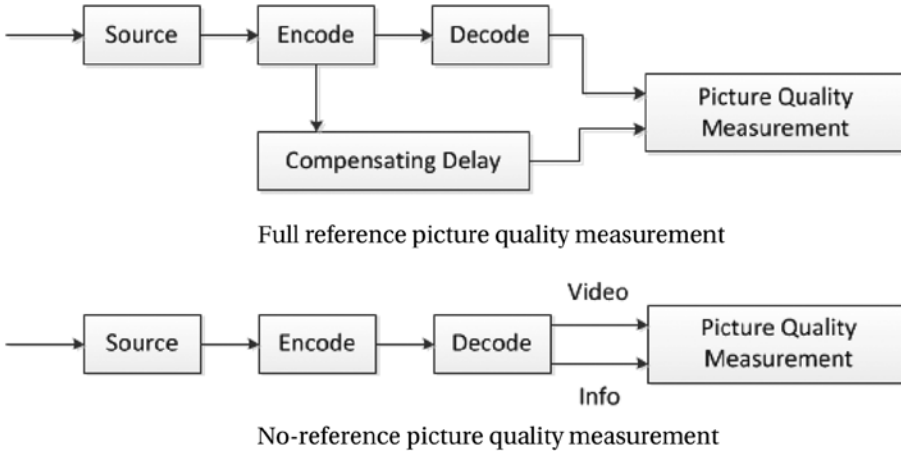


Figure 4-11. *Reference-based classification examples: FR and NR approaches*

An interesting NR quality measurement algorithm is presented in Wang et al.[8] This algorithm considers blurring and blocking as the most significant artifacts generated during the JPEG compression process, and proposes to extract features that can be used to reflect the relative magnitudes of these artifacts. The extracted features are combined to generate a quality prediction model that is trained using subjective experimental results. It is expected that the model would be a good fit for images outside the experimental set as well. While such algorithms are interesting in that an assessment can be made solely on the basis of the available image content without using any reference, it is always better to use an FR approach when the reference is available, so the following discussion will focus on the various FR approaches.

The problem of designing an objective metric that closely agrees with perceived visual quality under all conditions is a hard one. Many available metrics may not account for all types of distortion corrupting an image, or the content of the image, or the strength of the distortion, yet provide the same close agreement with human judgments. As such

[8]Z. Wang, H. R. Sheikh, and A. C. Bovik, "No-Reference Perceptual Quality Assessment of JPEG Compressed Images," *Proceedings of International Conference on Image Processing* 1, (2002): 477-80.

this remains an active research area. Several FR approaches have been taken in the quest for finding a good solution to this problem. Some of these approaches are presented in the following sections.

Error Sensitivity Based Approaches

When an image or video frame goes through lossy processing, a distorted image or video frame is produced. The amount of error or the distortion that is introduced by the lossy process determines the amount of visual quality degradation. Many quality-evaluation metrics are based on the error or intensity difference between the distorted image and the reference image pixels. The simplest and most widely used full-reference quality metric is the *mean squared error* (MSE), along with the related quantity of *peak signal-to-noise ratio* (PSNR). These are appealing because they are simple to calculate, have clear physical meanings, and are mathematically convenient in the context of optimization. However, they are not very well matched to perceived visual quality.

In error sensitivity based image or video quality assessment, it is generally assumed that the loss of perceptual quality is directly related to the visibility of the error signal. The simplest implementation of this concept is the MSE, which objectively quantifies the strength of the error signal. But two distorted images with the same MSE may have very different types of errors, some of which are much more visible than others. In the last four decades, a number of quality-assessment methods have been developed that exploit known characteristics of the human visual system (HVS). The majority of these models have followed a strategy of modifying the MSE measure so that different aspects of the error signal are weighted in accordance with their visibility. These models are based on a general framework, as discussed below.

General Framework

Figure 4-12 depicts a general framework of error sensitivity based approaches of image or video quality assessment. Although the details may differ, most error sensitivity based perceptual quality assessment models can be described with a similar block diagram.

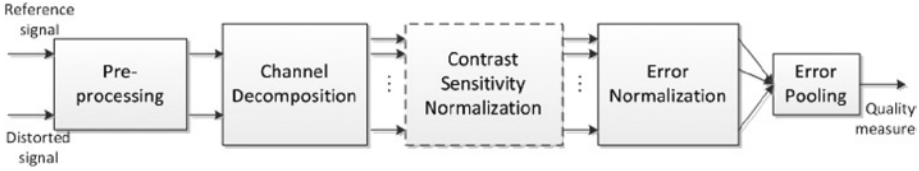


Figure 4-12. *General framework of error sensitivity based approaches*

In Figure 4-12, the general framework includes the following stages:

- **Pre-processing:** In this stage, known malformations are eliminated and the images are prepared so as to perform a fair comparison between the distorted image and the reference image. For example, both images are properly scaled and aligned. If necessary, a color space conversion or gamma correction may be performed that is more appropriate for the HVS. Further, a low-pass filter simulating the point spread function of the eye optics may be applied. Additionally, both images may be modified using a non-linear point operation to simulate light adaptation.
- **Channel Decomposition:** The images are typically separated into subbands or channels that are sensitive to particular spatial and temporal frequency, as well as orientation. Some complex methods try to closely simulate neural responses in primary visual cortex, while others simply use DCT or wavelet transforms for channel decomposition.
- **Contrast Sensitivity Normalization:** The *contrast sensitivity function* (CSF) describes the sensitivity of the HVS to different spatial and temporal frequencies that are present in the visual stimulus. The frequency response of the CSF is typically implemented as a linear filter. Some older methods weigh the signal according to CSF in a stage before channel decomposition, but recent methods use CSF as a base-sensitivity normalization factor for each channel.
- **Error Normalization:** The presence of one image component may decrease or mask the visibility of another, nearby image component, which is in close proximity in terms of spatial or temporal location, spatial frequency, or orientation. Such masking effect is taken into account when the error signal in each channel is calculated and normalized. The normalization process weighs the error signal in a channel by a space-varying visibility threshold. For each channel, the visibility threshold is determined based on the channel's base-sensitivity, as well as the energy of the reference or distorted image coefficients in a spatial neighborhood, either within the same channel or across channels. The normalization process expresses the error in terms of *just noticeable difference* (JND) units. Some methods also consider the effect of saturation of the contrast response.

- **Error Pooling:** In this final stage, the normalized error signals are combined to a single value. To obtain the combined value, typically the Minkowski norm is calculated as follows:

$$E(\{e_{i,j}\}) = \left(\sum_i \sum_j |e_{i,j}|^{\beta} \right)^{\frac{1}{\beta}} \quad \text{(Equation 4-2)}$$

  where $e_{i,j}$ is the normalized error of the j^{th} spatial coefficient in the i^{th} frequency channel, and β is a constant with typical values between 1 and 4.

Limitations

Although error sensitivity based approaches estimate the visibility of the error signal by simulating the functional properties of the HVS, most of these models are based on linear or quasilinear operators that have been characterized using restricted and simplistic stimuli. In practice, however, the HVS is a complex and highly non-linear system. Therefore, error sensitivity based approaches adopt some assumptions and generalizations leading to the following limitations:

- **Quality definition:** As error sensitivity based image or video quality assessment methods only track the image fidelity, lower fidelity does not always mean lower visual quality. The assumption that visibility of error signal translates to quality degradation may not always be valid. Some distortions are visible, but are not so objectionable. Brightening an entire image by globally increasing the luma value is one such example. Therefore, image fidelity only moderately correlates with image quality.
- **Generalization of models:** Many error-sensitivity models are based on experiments that estimate the threshold at which a stimulus is barely visible. These thresholds are used to define error-sensitivity measures such as the contrast sensitivity function. However, in typical image or video processing, perceptual distortion happens at a level much higher than the threshold. Generalization of near-threshold models in suprathreshold psychophysics is thus susceptible to inaccuracy.

- **Signal characteristics:** Most psychophysical experiments are conducted using relatively simple patterns, such as spots, bars, or sinusoidal gratings. For example, the CSF is typically obtained from threshold experiments using global sinusoidal images. However, real-world natural images have much different characteristics from the simple patterns. Therefore, the applicability of the simplistic models may be limited in practice.
- **Dependencies:** It is easy to challenge the assumption used in error pooling that error signals in different channels and spatial locations are independent. For linear channel decomposition methods such as the wavelet transform, a strong dependency exists between intra- and inter-channel wavelet coefficients of natural images. Optimal design of transformation and masking models can reduce both statistical and perceptual dependencies. However, the impact of such design on VQA models is yet to be determined.
- **Cognitive interaction:** It is well known that interactive visual processing such as eye movements influences the perceived quality. Also, cognitive understanding has a significant impact on quality. For example, with different instructions, a human subject may give different scores to the same image. Prior knowledge of or bias toward an image, attention, fixation, and so on may also affect the evaluation of the image quality. However, most error sensitivity based image or video quality assessment methods do not consider the cognitive interactions as they are not well understood and are difficult to quantify.

Peak Signal-to-Noise Ratio

The term *peak signal-to-noise ratio* (PSNR) is an expression for the ratio between the maximum possible power of a signal and the power of distorting noise that affects the quality of its representation after compression, processing, or transmission. Because many signals have a very wide *dynamic range* (ratio between the largest and smallest possible values of a changeable quantity), the PSNR is usually expressed in terms of the logarithmic decibel (dB) scale. The PSNR does not always perfectly correlate with a perceived visual quality, owing to the non-linear behavior of the HVS, but as long as the video content and the codec type are not changed, it is a valid quality measure,[9] as it is a good indicator of the fidelity of a video signal in a lossy environment.

Let us consider a signal f that goes through some processing or transmission and is reconstructed as an approximation $\hat{f}$, where some noise is introduced. Let f_m is the peak or maximum signal value; for n-bit representation of the signal $f_m = 2^n - 1$. For example, in

[9]Q. Huynh-Thu, and M. Ghanbari, "Scope of Validity of PSNR in Image/Video Quality Assessment," *Electronic Letters* 44, no. 13 (2008): 800–801.

case of an 8-bit signal $f_m = 255$, while for a 10-bit signal, $f_m = 1023$. PSNR, as a ratio of signal power to the noise power, is defined as follows:

$$PSNR = 10 \log_{10} \frac{(f_m)^2}{MSE} \quad \text{(Equation 4-3)}$$

where the mean square error (MSE) is given by:

$$MSE = \frac{1}{N} \sum_{i=1}^{N} (f_i - \hat{f}_i)^2 \quad \text{(Equation 4-4)}$$

where N is the number of samples over which the signal is approximated. Similarly, the MSE for a two-dimensional signal such as image or a video frame with width M and height N is given by:

$$MSE = \frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} (f(i,j) - \hat{f}(i,j))^2 \quad \text{(Equation 4-5)}$$

where $f(i, j)$ is the pixel value at location (i, j) of the source image, and $\hat{f}(i,j)$ is the corresponding pixel value in the reconstructed image. PSNR is usually measured for an image plane, such as the luma or chroma plane of a video frame.

Applications

PSNR has traditionally been used in analog audio-visual systems as a consistent quality metric. Digital video technology has exposed some limitations in using the PSNR as a quality metric. Nevertheless, owing to its low complexity and easy measurability, PSNR is still the most widely used video quality metric for evaluating lossy video compression or processing algorithms, particularly as a measure of gain in quality for a specified target bit rate for the compressed video. PSNR is also used in detecting the existence of frame drops or severe frame data corruption and the location of dropped or corrupt frames in a video sequence in automated environments. Such detections are very useful in debugging and optimization of video encoding or processing solutions. Furthermore, PSNR is extensively used as a comparison method between two video coding solutions in terms of video quality.

Advantages

PSNR has the following advantages as a video quality metric.

- PSNR is a simple and easy to calculate picture-based metric. PSNR calculation is also fast and parallelization-friendly—for example, using single instruction multiple data (SIMD) paradigm.
- Since PSNR is based on MSE, it is independent of the direction of the difference signal; either the source or the reconstructed signal can be subtracted from one another yielding the same PSNR output.

- PSNR is easy to incorporate into practical automated quality- measurement systems. This flexibility makes it amenable to a large test suite. Thus, it is very useful in building confidence on the evaluation.
- The PSNR calculation is repeatable; for the same source and reconstructed signals, the same output can always be obtained. Furthermore, PSNR does not depend on the width or height of the video, and works for any resolution.
- Unlike cumbersome subjective tests, PSNR does not require special setup for the environment.
- PSNR is considered to be a reference benchmark for developing various other objective video-quality metrics.
- For the same video source and the same codec, PSNR is a consistent quality indictor, so it can be used for encoder optimization to maximize the subjective video quality and/or the performance of the encoder.
- PSNR can be used separatelyfor luma and chroma channels. Thus, variation in brightness or color between two coding solutions can be easily tracked. In order to determine which solution uses more bits for a given quality level, such information is very useful.
- The popularity of PSNR is not only rooted in its simplicity but also its performance as a metric should not be underestimated. A validation study conducted by the Video Quality Experts Group (VQEG) in 2001 discovered that the nine VQA methods that it tested, including some of the most sophisticated algorithms at that time, were "statistically indistinguishable" from the PSNR.[10]

Limitations

Common criticisms for PSNR include the following.

- Some studies have shown that PSNR poorly correlates with subjective quality.[11]
- PSNR does not consider the visibility differences of two different images, but only considers the numerical differences. It does not take the visual masking phenomenon or the characteristics of the HVS into account–all pixels that are different in two images contribute to the PSNR, regardless of the visibility of the difference.

[10] *Final Report from the Video Quality Experts Group on the Validation of Objective Models of Video Quality Assessment,* 2000, available at www.vqeg.org.
[11] B. Girod, "What's Wrong with Mean Squared Error?" in *Visual Factors of Electronic Image Communications,* ed. A. B. Watson, (Cambridge, MA: MIT Press, 1993): 207–20.

- Other objective perceptual quality metrics have been shown to outperform PSNR in predicting subjective video quality in specific cases.[12]
- Computational complexity of the encoder in terms of execution time or machine cycles is not considered in PSNR. Nor does it consider system properties such as data cache size, memory access bandwidth, storage complexity, instruction cache size, parallelism, and pipelining, as all of these parameters contribute to coding complexity of an encoder. Therefore, the comparison of two encoders is quite restricted when PSNR is used as the main criteria.
- PSNR alone does not provide sufficient information regarding coding efficiency of an encoder; a corresponding cost measure is also required, typically in terms of the number of bits used. In other words, saying that a certain video has a certain level of PSNR does not make sense unless the file size or the bit rate for the video is also known.
- PSNR is typically averaged over a frame, and local statistics within the frame are not considered. Also, for a video sequence, the quality may vary considerably from scene to scene, which may not be accurately captured if frame-based PSNR results are aggregated and an averge PSNR is used for the video sequence.
- PSNR does not capture temporal quality issues such as frame delay or frame drops. Additionally, PSNR is only a source coding measure and does not consider channel coding issues such as multi-path propagation or fading. Therefore, it is not a suitable quality measure in lossy network environment.
- PSNR is an FR measure, so reference is required for quality evaluation of a video. However, in practice, an unadulterated reference is not generally available at the reconstruction end. Nevertheless, PSNR remains effective and popular for evaluation, analysis, and benchmarking of video quality.

Improvements on PSNR

Several attempts have been made in literature to improve PSNR. Note that visibility of a given distortion depends on the local content of the source picture. Distortions are usually more objectionable in plain areas and on edges than in *busy* areas. Thus, it is possible to model the visual effect of the distortion itself in a more sophisticated way than

[12] *ITU-T Recommendation J.144: Objective Perceptual Video Quality Measurement Techniques for Digital Cable Television in the Presence of a Full Feference* (Geneva, Switzerland: International Telecommunications Union, 2004).

simply measuring its energy, as done in PSNR. For example, a weighting function may be applied in frequency domain, giving more weight to the lower-frequency components of the error than to the higher-frequency components. A new measure, named *just noticeable difference* (JND), has been defined by Sarnoff in 2003, based on a visual discrimination mode.[13]

Moving Picture Quality Metric

PSNR does not take the visual masking phenomenon into consideration. In other words, every single pixel error contributes to the decrease of the PSNR, even if this error is not perceptible. This issue is typically addressed by incorporating some HVS models. In particular, two key aspects of the HVS, namely contrast sensitivity and masking, have been intensively studied in the literature. The first phenomenon accounts for the fact that a signal is detected by the eye only if its contrast is greater than some threshold. The sensitivity of the eye varies as a function of spatial frequency, orientation, and temporal frequency. The second phenomenon is related to the human vision response to a combination of several signals. For example, consider a stimulus consisting of the foreground and the background signals. The detection threshold of the foreground is modified as a function of the contrast from the background.

The *moving picture quality metric* (MPQM)[14] is an error-sensitivity based spatio-temporal objective quality metric for moving pictures that incorporates the two HVS characteristics mentioned above. Following the general framework shown in Figure 4-12, MPQM first decomposes an original video and a distorted version of it into perceptual channels. A channel-based distortion measure is then computed, accounting for contrast sensitivity and masking. After obtaining the distortion data for each channel, the data is combined over all the channels to compute the quality rating. The resulting quality rating is then scaled from1 to 5 (from bad to excellent). MPQM is known to give good correlation with subjective tests for some videos, but it also yields bad results for others.[15] This is consistent with other error-sensitivity based approaches.

The original MPQM algorithm does not take chroma into consideration, so a variant of the algorithm called the *color MPQM* (CMPQM) has been introduced. In this technique, first the color components are converted to RGB values that are linear with luminance. Then the RGB values are converted to coordinate values corresponding to a luma and two chroma channels. This is followed by the analysis of each component of the original and error sequence by a filter bank. As the HVS is less sensitive to chroma, only nine spatial and one temporal filter is used for these signals. The rest of the steps are similar to those in MPQM.

---

[13]J. Lubin, "A Visual Discrimination Mode for Image System Design and Evaluation," in *Visual Models for Target Detection and Recognition,* ed. E. Peli, (Singapore: World Scientific Publishers, 1995): 207–20.

[14]C. J. Branden-Lambrecht, and O. Verscheure, "Perceptual Quality Measure using a Spatio-Temporal Model of the Human Visual System," in *Proceedings of the SPIE* 2668 (San Jose, CA: SPIE-IS&T, 1996): 450–61.

[15]See `http://www.irisa.fr/armor/lesmembres/Mohamed/Thesis.pdf`.

Structural Similarity Based Approaches

Natural image signals are highly structured. There are strong dependencies among the pixels of natural images, especially when they are spatially adjacent. These dependencies carry important information about the structure of the objects in a visual scene. The Minkowski error metric used in error sensitivity based approaches does not consider the underlying structure of the signal. Also, decomposition of image signal using linear transforms, as done by most quality measures based on error sensitivity, do not remove the strong dependencies. Structural similarity based quality assessment approaches try to find a more direct way to compare the structures of the reference and the distorted signals. Based on the HVS characteristic that human vision reacts quickly to structural information in the viewing field, these approaches approximate the perceived image distortion using a measure of structural information change. The *Universal Image Quality Index* (UIQI)[16] and the ***S**tructural Similarity Index* (SSIM) [17] are two examples of this category. For a deeper understanding, the SSIM is discussed below in detail.

Structural Similarity Index

Objective methods for assessing perceptual image quality attempt to measure the visible differences between a distorted image and a reference image using a variety of known properties of the HVS. Under the assumption that human visual perception is highly adapted for extracting structural information from a scene, a quality assessment method was introduced based on the degradation of structural information.

The structural information in an image is defined as those attributes that represent the structure of objects in a scene, independent of the luminance and contrast. Since luminance and contrast can vary across a scene, structural similarity index (SSIM) analysis only considers the local luminance and contrast. As these three components are relatively independent, a change in luminance or contrast of an image would not affect the structure of the image.

The system block diagram of the structural similarity index based quality assessment system is shown in Figure 4-13.

[16]Z. Wang, and A. C. Bovik, "A Universal Image Quality Index," *IEEE Signal Processing Letters* 9, no. 3 (March 2002): 81–84.

[17]Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image Quality Assessment: From Error Visibility to Structural Similarity," *IEEE Transactions on Image Processing* 13, no. 4 (April 2004): 600–12.

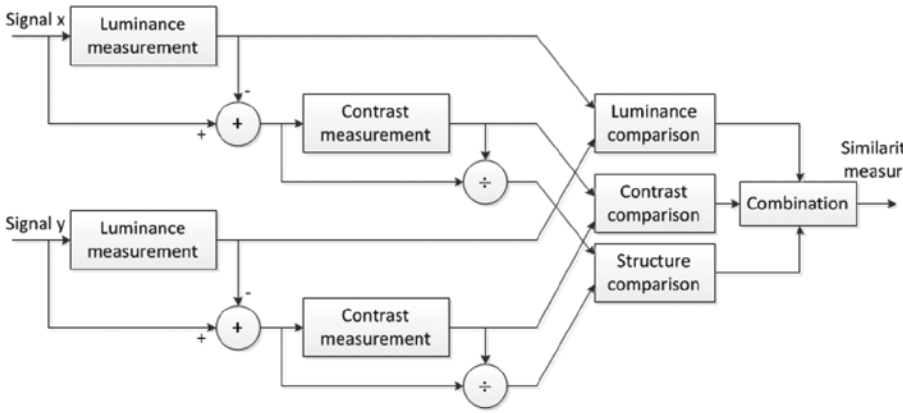


Figure 4-13. *Block diagram of the SSIM measurement system*

As shown in Figure 4-13, the system consists of two nonnegative spatially aligned image signals *x* and *y*. If one of the signals has perfect quality, then the similarity measure can serve as a quantitative measurement of the quality of the second signal. The system separates the task of similarity measurement into three comparisons: luminance, contrast, and structure.

Luminance is estimated as the mean intensity (μ) of the image. The luminance comparison function $l(x,y)$ is a function of the mean intensities μ_x and μ_y of images *x* and *y*, and can be obtained by comparing their mean intensities. Contrast is estimated as the standard deviation (σ) of an image. The contrast comparison function $c(x,y)$ then reduces to a comparison of σ_x and σ_y. In order to perform the structure comparison, the image is first normalized by dividing the singal by its own standard deviation, so that both images have unit standard deviation. The structure comparison $s(x,y)$ is then done on these normalized signals $(x-\mu_x)/\sigma_x$ and $(y-\mu_y)/\sigma_y$. Combining the results of these three comparisons yields an overall similarity measure::

$$S(x,y) = f(l(x,y), c(x,y), s(x,y)). \qquad \text{(Equation 4-6)}$$

The similarity measure is designed to satisfy the following conditions:

- Symmetry: $S(x,y) = S(y,x)$
- Boundedness: $S(x,y) \le 1$
- Unity maximum: $S(x,y) = 1$ if and only if $x = y$ (in discrete representations, $x_i = y_i, \forall i = 1, \cdots, N$.)

The luminance comparison function is defined as):

$$l(x,y) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1} \qquad \text{(Equation 4-7)}$$

Here, the constant C_1 is introduced to avoid instability when $\mu_x^2+\mu_y^2$ is close to zero. Specifically, C_1 is chosen as: $C_1=(K_1L)^2$, where L is the dynamic range of the pixel values (e.g., 255 for 8-bit grayscale pixels), and the constant $K_1 \ll 1$ is a small constant. Qualitatively, Equation 4-7 is consistent with Weber's law, which is widely used to model light adaptation or luminance masking in the HVS. In simple terms, Weber's law states that the HVS is sensitive to the relative luminance change, and not to the absolute luminance change. If R represents the relative luminance change compared to the background luminance, the distorted signal mean intensity can be substituted by $\mu_y=(1+R)\mu_x$ and Equation 4-7 can be rewritten as ():

$$l(\boldsymbol{x},\boldsymbol{y})=\frac{2(1+R)}{1+(1+R)^2+\frac{C_1}{\mu_x^2}} \qquad \text{(Equation 4-8)}$$

For small values of C_1 with respect to μ_x^2, $l(x,y)=f(R)$, which is consistent with Weber's law.

The contrast comparison function takes a similar form ():

$$c(\boldsymbol{x},\boldsymbol{y})=\frac{2\sigma_x\sigma_y+C_2}{\sigma_x^2+\sigma_y^2+C_2} \qquad \text{(Equation 4-9)}$$

where $C_2=(K_2L)^2$, and $K_2 \ll 1$. Note that with the same amount of contrast change $\Delta\sigma=\sigma_y-\sigma_x$, this measure is less sensitive to a high-base contrast than a low-base contrast. This is consistent with the contrast-masking feature of the HVS.

Structure comparison is conducted after luminance subtraction and variance normalization. Specifically, the two unit vectors $(x-\mu_x)/\sigma_x$ and $(y-\mu_y)/\sigma_y$, are associated with the structure of the two images. The correlation between these two vectors can simply and effectively quantify the structural similarity. Notice that the correlation between $(x-\mu_x)/\sigma_x$ and $(y-\mu_y)/\sigma_y$ is equivalent to the correlation coefficient between $\boldsymbol{x}$ and $\boldsymbol{y}$. Thus, the structure comparison function is defined as follows:

$$s(x,y)=\frac{\sigma_{xy}+C_3}{\sigma_x\sigma_y+C_3} \qquad \text{(Equation 4-10)}$$

As in the luminance and contrast measures, a small constant C_3 is introduced for stability. In discrete form, σ_{xy}can be estimated as :

$$\sigma_{xy}=\frac{1}{N-1}\sum_{i=1}^{N}(x_i-\mu_x)(y_i-\mu_y) \qquad \text{(Equation 4-11)}$$

The three comparisons of Equations 4-8, 4-9, and 4-10 are combined to yield the resulting similarity measure SSIM between signals x and y:

$$SSIM(\boldsymbol{x},\boldsymbol{y})=[l(x,y)]^{\alpha}[c(x,y)]^{\beta}[s(x,y)]^{\gamma} \qquad \text{(Equation 4-12)}$$

where $\alpha>0$, $\beta>0$, and $\gamma>0$ are parameters used to adjust the relative importance of the three components.

The expression is typically used in a simplified form, with $\alpha=\beta=\gamma=1$ and $C_3=C_2/2$:

$$SSIM(\boldsymbol{x},\boldsymbol{y})=\frac{(2\mu_x\mu_y+C_1)(2\sigma_{xy}+C_2)}{(\mu_x^2+\mu_y^2+C_1)(\sigma_x^2+\sigma_y^2+C_2)} \qquad \text{(Equation 4-13)}$$

The UIQI[6] is a special case of SSIM with $C_1=C_2=0$. However, it produces unstable results when either $(\mu_x{}^2+\mu_y{}^2)$ or $(\sigma_x{}^2+\sigma_y{}^2)$ is very close to zero.

Information Fidelity Based Approaches

Images and videos generally involve natural scenes, which are characterized using statistical models. Most real-world distortion processes disturb these statistics and make the image or video signals unnatural. This observation led researchers to use *natural scene statistics* (NSS) models in conjunction with a distortion (channel) model to quantify the information shared between a distorted and a reference image, and to show that this shared information is an aspect of signal fidelity that relates well with visual quality. Although in contrast to the HVS error-sensitivity and the structural approaches, the statistical approach, as used in an information-theoretic setting, does not rely on any HVS parameter, or constants requiring optimization, it still yields an FR QA method that is competitive with state-of-the-art QA methods. The visual information fidelity (VIF) is such an information-fidelity based video quality assessment metric.

Visual Information Fidelity

Visual Information Fidelity[18] (VIF) is an information theoretic criterion for image fidelity measurement based on NSS. The VIF measure quantifies the information that could ideally be extracted by the brain from the reference image. Then, the loss of this information to the distortion is quantified using NSS, HVS and an image distortion (channel) model in an information-theoretic framework. It was found that visual quality of images is strongly related to relative image information present in the distorted image, and that this approach outperforms state-of-the-art quality-assessment algorithms. Further, VIF is characterized by only one HVS parameter that is easy to train and optimize for improved performance.

VIF utilizes NSS models for FR quality assessment, and models natural images in the wavelet domain using the well-known Gaussian Scale Mixtures (GSM). Wavelet analysis of images is useful for natural image modeling. The GSM model has been shown to capture key statistical features of natural images, such as linear dependencies in natural images.

Natural images of perfect quality can be modeled as the output of a stochastic source. In the absence of any distortions, this signal passes through the HVS before entering the brain, which extracts cognitive information from it. For distorted images, it is assumed that the reference signal has passed through another *distortion channel* before entering the HVS.

[18]H. R. Sheikh and A. C. Bovik, "Image Information and Visual Quality," *IEEE Transactions on Image Processing* 15, no. 2 (2006): 430–44.

The distortion model captures important, and complementary, distortion types: blur, additive noise, and global or local contrast changes. It assumes that in terms of their *perceptual annoyance*, real-world distortions could roughly be approximated locally as a combination of blur and additive noise. A good distortion model is one where the distorted image and the synthesized image look equally perceptually annoying, and the goal of the distortion model is not to model image artifacts but the perceptual annoyance of the artifacts. Thus, even though the distortion model may not be able to capture distortions such as ringing or blocking exactly, it may still be able to capture their perceptual annoyance. However, for distortions other than blur and white noise— for example, for low-bit-rate compression noise—the model fails to adequately reproduce the perceptual annoyance.

The HVS model is also described in the wavelet domain. Since HVS models are duals of NSS models, many aspects of HVS are already captured in the NSS description, including wavelet channel decomposition, response exponent, and masking effect modeling. In VIF, the HVS is considered a distortion channel that limits the amount of information flowing through it. All sources of HVS uncertainty are lumped into one additive white Gaussian stationary noise called the *visual noise*.

The VIF defines mutual informations $I(\vec{C}^N;\vec{E}^N \mid s^N)$ and $I(\vec{C}^N;\vec{F}^N \mid s^N)$ to be the information that could ideally be extracted by the brain from a particular subband in the reference and the distorted images, respectively. Intuitively, visual quality should relate to the amount of image information that the brain could extract from the distorted image relative to the amount of information that the brain could extract from the reference image. For example, if the brain can extract 2.0 bits per pixel of information from the distorted image when it can extract 2.1 bits per pixel from the reference image, then most of the information has been retrieved and the corresponding visual quality should be very good. By contrast, if the brain can extract 5.0 bits per pixel from the reference image, then 3.0 bits per pixel information has been lost and the corresponding visual quality should be very poor.

The VIF is given by:

$$VIF = \frac{\sum_{j \in subbands} I(\vec{C}^{N,j};\vec{F}^{N,j} \mid s^{N,j})}{\sum_{j \in subbands} I(\vec{C}^{N,j};\vec{E}^{N,j} \mid s^{N,j})} \qquad \text{(Equation 4-14)}$$

where the sum is performed over the subbands of interest, and $\vec{C}^{N,j}$ represent N elements of the random field *Cj* that describes the coefficients from subband *j*, and so on.

The VIF has many interesting properties. For example, VIF is bounded below by zero, indicating all information is lost in the distortion channel. If a test image is just a copy of itself, it is not distorted at all, so the VIF is unity. Thus, VIF is always in the range [0,1]. Interestingly, a linear contrast enhancement of the reference image that does not add noise to it will result in a VIF value larger than unity, thereby signifying that the enhanced image has a superior visual quality to the reference image. This is a unique property not exhibited by other VQA metrics.

Spatio-Temporal Approaches

Traditional FR objective quality metrics do not correlate well with temporal distortions such as frame drops or jitter. Spatio-temporal approaches are more suitable for video signals as they consider the motion information between video frames, thereby capturing temporal quality degradation as well. As a result, these algorithms generally correlate well with the HVS. As an example of this approach, the spatio-temporal video SSIM (stVSSIM) is described.

Spatio-Temporal Video SSIM

Spatio-temporal video SSIM[19] (stVSSIM) algorithm is a full-reference VQA algorithm based on the *motion-based video integrity evaluation*[20] (MOVIE) algorithm. MOVIE utilizes a multi-scale spatio-temporal Gabor filter bank to decompose the videos and to compute motion vectors. However, MOVIE has high computational complexity, making practical implementations difficult. So, stVSSIM proposes a new spatio-temporal metric to address the complexity issue. The stVSSIM algorithm was evaluated on VQEG's full-reference data set (Phase I for 525 and 625 line TV signals) and was shown to perform extremely well in terms of correlation with human perception.

For spatial quality assessment, stVSSIM uses the single-scale structural similarity index (SS-SSIM) as it correlates well with human perception of visual quality. For temporal quality assessment, stVSSIM extends the SS-SSIM to the spatio-temporal domain and calls it SSIM-3D. Motion information is incorporated in the stVSSIM using a block-based motion estimation algorithm, as opposed to optical flow, as used in MOVIE. Further, a method to completely avoid block motion estimation is introduced, thereby reducing computational complexity.

For spatial quality assessment, SS-SSIM is computed on a frame-by-frame basis. The spatial-quality measure is applied on each frame and the frame-quality measure is computed using the percentile approach. As humans tend to rate images with low-quality regions with greater severity, using a percentile approach would enhance algorithm performance. So, *Percentile-SSIM* or *P-SSIM* is applied on the scores obtained for each frame. Specifically, the frame-quality measure is:

$$S_{frame} = \frac{1}{|\varphi|}\sum_{i\in\varphi} SSIM(i) \qquad \text{(Equation 4-15)}$$

where the set of the lowest 6 percent of SSIM values from the frame and SSIM(i) the SS-SSIM score is at pixel location i.

The spatial score for the video is computed as the mean of the frame-level scores and is denoted as S_{video}.

[19]A. K. Moorthy and A. C. Bovik, "Efficient Motion Weighted Spatio-Temporal Video SSIM Index," in *Proceedings of SPIE-IS&T Electronic Imaging* 7527 (San Jose, CA: SPIE-IS&T, 2010): 1–9.
[20]K. Seshadrinathan and A. C. Bovik, "Motion-based Perceptual Quality Assessment of Video," in *Proceedings of the SPIE* 7240 (San Jose, CA: SPIE-IS&T, 2009): 1–12.

Temporal quality evaluation utilizes three-dimensional structural similarity (SSIM-3D) for a section of the video and performs a weighting on the resulting scores using motion information derived from motion vectors. In this case, a video is viewed as a three-dimensional signal. If x and y are the reference and the distorted video, a volume section is defined around a pixel location (i,j,k) with spatial dimensions (α, β) while the volume temporally encompasses γ frames. Here, (i,j) correspond to the spatial location and k corresponds to the frame number. The SSIM-3D is then expressed as a 3-D extension of the SSIM as follows:

$$SSIM_{3D} = \frac{(2\mu_{x(i,j,k)}\mu_{y(i,j,k)} + C_1)(2\sigma_{x(i,j,k)y(i,j,k)} + C_2)}{(\mu^2_{x(i,j,k)} + \mu^2_{y(i,j,k)} + C_1)(\sigma^2_{x(i,j,k)} + \sigma^2_{y(i,j,k)} + C_2)} \quad \text{(Equation 4-16)}$$

To compute the 3-D mean μ, the variance σ^2, and the co-variance σ_{xy}, the sections x and y are weighted with a weighting factor w for each dimension (i,j,k). The essence of stVSSIM is evaluating spatio-temporal quality along various orientations at a pixel, followed by a weighting scheme that assigns a spatio-temporal quality index to that pixel. The weighting factor depends on the type of filter being used—one out of the four proposed spatio-temporal filters (vertical, horizontal, left, and right).

To incorporate motion information, block motion estimation is used, where motion vectors are computed between neighboring frames using the Adaptive Rood Pattern Search (ARPS) algorithm operating on 8×8 blocks. Once motion vectors for each pixel (i,j,k) are available, spatio-temporal SSIM-3D scores are weighted. To avoid weighting that uses floating point numbers, a greedy weighting is performed. In particular, the spatio-temporal score at pixel (i,j,k) is selected from the scores produced by the four filters based on the type of filter that is closest to the direction of motion at pixel (i,j,k). For example, if the motion vector at a pixel were $(u,v) = (0,2)$, the spatio-temporal score of that pixel would be the SSIM-3D value produced by the vertical filter. If the motion vector is equidistant from two of the filter planes, the spatio-temporal score is the mean of the SSIM-3D scores of the two filters. In case of zero motion, the spatio-temporal score is the mean of all four SSIM-3D values.

The temporal score for the video is computed as the mean of the frame-level scores and is denoted as T_{video}. The final score for the video is given by $S_{video} \times T_{video}$.

Saliency Based Approaches

Quality-assessment methods suitable for single images are also typically used for video. However, these methods do not consider the motion information of the video sequence. As a result, they turn out to be poor evaluation metrics for video quality. In addition, most VQA algorithms ignore the human visual attention mechanism, which is an important HVS characteristic.

Human eyes usually focus on edges with high-contrast or *salient* areas that are different from their neighboring areas. Recognizing this fact, saliency based approaches of video quality evaluation treat the distortion occuring in the salient areas asymmetrically compared to that occuring in other areas. One such approach is SVQA.[21]

[21]Q. Ma, L. Zhang, and B. Wang, "New Strategy for Image and Video Quality Assessment," *Journal of Electronic Imaging* 19, no. 1 (2010): 1–14.

Saliency-based Video Quality Assessment

In a saliency based video quality-assessment (SVQA) approach, a spatial saliency map is extracted from reference images or video frames using a fast frequency domain method called the *phase spectrum of quaternion Fourier transform* (PQFT). When the inverse Fourier transform is taken of an image phase spectrum, salient areas are easily recognizable. The saliency map is used as weights to adjust other objective VQA criteria, such as PSNR, MSSIM, VIF, and so on. Similarly, temporal weights are determined from adjacent frames.

Given a reference image and a corresponding distorted image, the saliency map of the reference image can be obtained using the PQFT. Then, an improved quality assessment index, called the saliency-based index (S-index), is determined by weighting the original index by the saliency map. For example, if p_i is the luma value of the i^{th} pixel in the salient area, the pixel saliency weight w_i is given by the following:

$$w_i = \frac{p_i + b}{\frac{1}{M \times N}\sum_{i=1}^{M \times N}(p_i + b)} \qquad \text{(Equation 4-17)}$$

where b is a small constant to keep $w_i > 0$, and M and N are the width and height of the image, respectively. Therefore, this weighting takes into account the non-salient areas as well. However, pixels in the salient area have large weights. Using these weights, the *saliency-based PSNR* (SPSNR) is written as:

$$SPSNR(x,y) = 10\log\frac{255^2}{SMSE(x,y)},$$
$$SMSE(x,y) = \frac{1}{M \times N}\sum_{i=1}^{M \times N}(x_i - y_i)^2 w_i. \qquad \text{(Equation 4-18)}$$

Thus, the distortion of pixels in the salient area is given more importance than pixels in other areas. *Saliency-based MSSIM* (SMSSIM) and *saliency-based VIF* (SVIF) are also defined in a similar manner.

As SVQA deals with video signals instead of images only, the following considerations are taken into account:

- HVS is sensitive to motion information, but less sensitive to the background. Therefore, distortion of moving objects is very important. SVQA differentiates between a fixed camera and a moving camera while locating a moving object.
- As frames are played out in real time, human eyes can only pay attention to a much smaller area in an image, compared to when looking at a fixed image. This is considered in intraframe weights.
- Due to *motion masking* effect, visual sensitivity is depressed during large-scale scene changes or rapid motion of objects. Therefore, frames should be weighted differently based on motion masking. This is considered in interframe weights.
- Considering spatio-temporal properties of video sequences, saliency weights in both spatial and temporal domains contribute to the final quality index.

In SVQA, the intraframe weight uses PQFT to calculate the saliency map for pixels with non-zero motion. Non-zero motion is represented as the difference image between two adjacent video frames. This establishes the first consideration of moving objects. In a short interval, the saliency map is allowed to have a square area of processing, thus addressing the second consideration. As the interframe weight is based on motion masking, the third consiration for weighting is also addressed. Finally, it is noteworthy that both intraframe weight and interframe weight are considered together in SVQA. Figure 4-14 shows the SVQA framework.

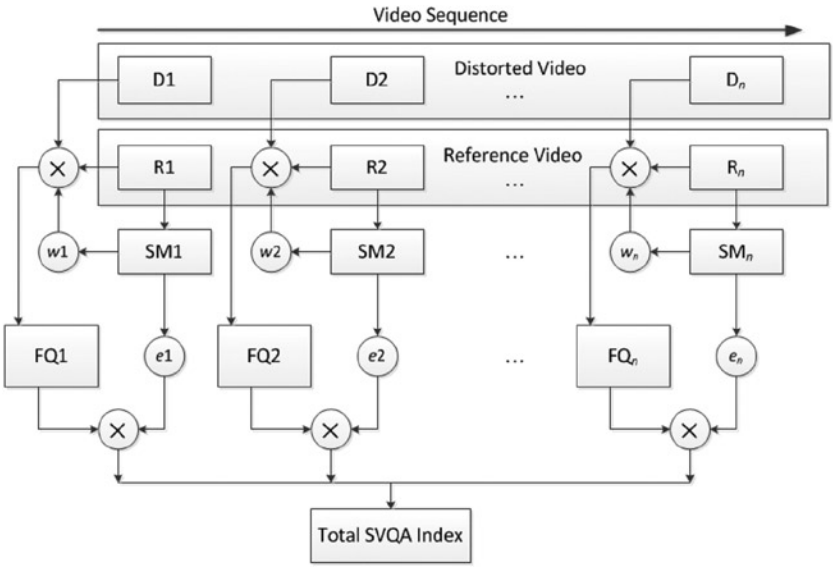


Figure 4-14. *The SVQA framework*

According to Figure 4-14, the flow of SVQA is as follows:

1. The reference and distorted video sequences are divided into frames. Each sequence is composed of n images: D_1 to D_n for distorted frames, R_1 to R_n for reference frames, as shown on the upper part of Figure 4-14.

2. Considering camera shift, where all pixels are moving, a binary function is defined to detect such motion. If the luma difference between pixels co-located in adjacent frames exceeds a threshold, a movement is detected. If the movement is detected for all pixels, a camera movement is understood; otherwise, object movement is considered on a static background. The quaternion for a frame is constructed as a weighted combination of motion information and three color channel information.

3. The *saliency map* (SM) is calculated for each reference video frame using the PQFT with motion, denoted as SM_1 to SM_n.
4. Based on the SM, the intraframe weights (w_1, ..., w_n) are calculated for n frames.
5. The frame quality FQ_i is calculated for the i^{th} frame using any of the saliency-based metrics such as SPSNR, SMSSIM, or SVIF.
6. Based on the SM, the interframe weights (e_1, ..., e_n) are calculated for n frames.
7. The SVQA index, the measure of quality of the entire video is calculated using the following equation:

$$SVQA = \frac{\sum_{i=1}^{n} FQ_i e_i}{\sum_{i=1}^{n} e_i} \qquad \text{(Equation 4-19)}$$

where n is the number of video frames.

Network-Aware Approaches

The objective video quality metrics such as PSNR neither perfectly correlate with perceived visual quality nor take the packet loss into account in lossy network environments such as multihop wireless mesh networks. While PSNR and similar metrics may work well for evaluating video quality in desktop coding applications and streaming over wired networks, remarkable inaccuracy arises when they are used to evaluate video quality over wireless networks.

For instance, in a wireless environment, it could happen that a video stream with a PSNR around 38dB (typically considered medium-high quality in desktop video coding applications) is actually perceived to have the same quality as the original undistorted video. This is because wireless video applications typically use the *User Datagram Protocol* (UDP), which does not guarantee reliable transmissions and may trade packet loss for satisfying delay requirements. Generally, in *wireless local area networks* (WLAN) consisting of unstable wireless channels, the probability of a packet loss is much higher than that in wired networks. In such environments, losing consecutive packets may cause the loss of an entire frame, thereby degrading the perceived video quality further than in desktop video coding applications.

Modified PSNR

Aiming to handle video frame losses, *Modified PSNR* (MPSNR) was proposed.[22] Two objective metrics are derived based on linear regression of PSNR against subjective MOS.

[22]A. Chan, K. Zeng, P. Mohapatra, S.-J. Lee, and S. Banerjee, "Metrics for Evaluating Video Streaming Quality in Lossy IEEE 802.11 Wireless Networks," *Proceedings of IEEE INFOCOM,* (San Diego, CA: March 2010): 1–9.

The first metric, called *PSNR-based Objective MOS* (POMOS), predicts the MOS from the mean PSNR, while achieving a correlation of 0.87 with the MOS. The second metric, called *Rate-based Objective MOS* (ROMOS), adds streaming network parameters such as the frame loss rate, and achieves a higher correlation of 0.94 with the MOS.

Frame losses are prevalent in wireless networks, but are not accounted for in the traditional PSNR calculations. Due to packet losses during streaming, a frame can be missing, which is typically unrecognizable by a human viewer. However, a missing frame causes the wrong frame to be compared against the original frame during PSNR calculation. Such off-position comparisons result in low PSNR values. A straightforward way to fix this is to introduce timing information into the source video. But such modification of source video is undesirable.

To determine if any frame is missing, an alternative approach is to match the frame with the original frame. The algorithm assumes that the sum of PSNRs of all frames is maximized when all frames are matching, and it uses this sum to determine the mismatching frame. In particular, MPSNR matches each frame in a streamed video to a frame in the reference video so that the sum of PSNR of all frame pairs is maximized. A moving window is used to determine the location of the matching frame. If frame j in the streamed video matches frame k belonging to the window in the reference video, it is considered that the frames (k-j) are missing. A frame in the streamed video need only be compared with, at most, g frames in the reference video, where g is the number of frames lost.

In addition to PSNR, the MPSNR measures the following video streaming parameters:

- Distorted frame rate (d): the percentage of mismatched frames in a streaming video.
- Distorted frame PSNR ($dPSNR$): the mean PSNR value of all the mismatched frames.
- Frame loss rate (l): the percentage of lost frames in a streaming video. It is calculated by comparing the total number of frames in the received streamed video with that in the reference video.

Once the corresponding frames in a streamed video and the reference video are matched, and the PSNR of each frame in the streamed video is calculated, all the above parameters are readily available.

In the MPSNR model, this method of matching is applied to a training set of videos, and the average PSNR for a window W is calculated. Experimental results show that the average PSNR exhibits a linear relationship with subjective MOS. Therefore, a linear model of the average PSNR can be used to predict the MOS score. The linear model is given as:

$$POMOS = 0.8311 + 0.0392\,(average\ PSNR) \qquad \text{(Equation 4-20)}$$

Note that average PSNR is used in this model. Since the average PSNR of a perfectly matching frame is infinity (or a very high value), it affects the prediction of MOS. To mitigate this problem, another linear model is proposed that does not use the PSNR values:

$$ROMOS = 4.367 - 0.5040\frac{d}{dPSNR} - 0.0517l. \qquad \text{(Equation 4-21)}$$

Noise-Based Quality Metrics

An interesting approach to quality evaluation is to evaluate the noise introduced instead of the signal fidelity.

Noise Quality Measure

In the *noise quality measure*[23] (NQM), a degraded image is modeled as an original image that has been subjected to linear frequency distortion and additive noise injection. These two sources of degradation are considered independent and are decoupled into two quality measures: a distortion measure (DM) resulting from the effect of frequency distortion, and a noise quality measure (NQM) resulting from the effect of additive noise.

The NQM is based on a contrast pyramid and takes into account the following:

- The variation in contrast sensitivity with distance, image dimensions, and spatial frequency
- The variation in the local brightness mean
- The contrast interaction between spatial frequencies
- The contrast masking effects

For additive noise, the non-linear NQM is found to be a better measure of visual quality than the PSNR and linear quality measures.

The DM is computed in three steps. First, the frequency distortion in the degraded image is found. Second, the deviation of this frequency distortion from an all-pass response of unity gain (no distortion) is computed. Finally, the deviation is weighted by a model of the frequency response of the HVS, and the resulting weighted deviation is integrated over the visible frequencies.

Objective Coding Efficiency Metrics

Measuring coding efficiency is another way to look at the tradeoff between visual quality and bit-rate cost in video coding applications. In this section we discuss the popular BD metrics for objective determination of coding efficiency.

[23]N. Damera-Venkata, T. D. Kite, W. S. Geisler, B. L. Evans, and A. C. Bovik, "Image Quality Assessment Based on a Degradation Model," *IEEE Transactions on Image Processing* 9, no. 4 (April 2000): 636–50.

BD-PSNR, BD-SSIM, BD-Bitrate

Bjøntegaard delta PSNR[24] (BD-PSNR) is an objective measure of coding efficiency of an encoder with respect to a reference encoder. It was proposed by Gisle Bjøntegaard in April 2001 in the Video Coding Expert Group's (VCEG) meeting. BD-PSNR considers the relative differences between two encoding solutions in terms of number of bits used to achieve a certain quality. In particular, BD-PSNR calculates the average PSNR difference between two rate-distortion (R-D) curves over an interval. This metric is a good indication of visual quality of encoded video, as it considers the cost (i.e., bits used) to achieve a certain visual quality of the decoded video, represented by the popular objective measure PSNR. Improvements to the BD-PSNR model can be performed by using $\log_{10}(bitrate)$ instead of simply the bit rate when plotting R-D data points, resulting in *straighter* R-D curves and more uniformly spaced data points across the axes.

BD-PSNR uses a third order logarithmic polynomial to approximate a given R-D curve. The reconstructed distortion in PSNR is given as:

$$D_{PSNR} = D(r) = a + br + cr^2 + dr^3 \qquad \text{(Equation 4-22)}$$

where $r = \log(R)$, R is the output bit rate, and a, b, c, and d are fitting parameters.

This model is a good fit to R-D curves and there is no problem with singular points, as could have happened for a model with $(r + d)$ in the denominator. The above equation can be solved with four R-D data points obtained from actual encoding, and the fitting parameters $a, b, c,$ and d can be determined. Thus, this equation can be used to interpolate the two R-D curves from the two encoding solutions, and the delta PSNR between the two curves can be obtained as:

$$BD\ PSNR = \frac{1}{(r_H - r_L)} \int_{r_L}^{r_H} (D_2(r) - D_1(r))dr \qquad \text{(Equation 4-23)}$$

where $r_H = \log(R_H)$, $r_L = \log(R_L)$ are the high and low ends, respectively, of the output bit rate range, and $D_1(r)$ and $D_2(r)$ are the two R-D curves.

Similarly, the interpolation can also be done on the bit rate as a function of SNR:

$$r = a + bD + cD^2 + dD^3 \qquad \text{(Equation 4-24)}$$

where $r = \log(R)$, R is the output bit rate, $a, b, c,$ and d are fitting parameters, and D is the distortion in terms of PSNR. From this the BD-bit rate can be calculated in a similar fashion as is done for PSNR above:

$$BD\ Bit\ rate = \frac{1}{D_H - D_L} \int_{D_L}^{D_H} (r_2 - r_1)dD \qquad \text{(Equation 4-25)}$$

[24]B. Bjøntegaard, *Calculation of Average PSNR Differences between RD curves (VCEG-M33)* (Austin, TX: ITU-T VCEG SG16 Q.6, 2001).

Therefore, from BD-PSNR calculations, both of the following can be obtained:

- Average PSNR difference in dB over the whole range of bit rates
- Average bit rate difference in percent over the whole range of PSNR

If the distortion measure is expressed in terms of SSIM instead of PSNR, BD-SSIM can be obtained in the same manner. BD-PSNR/BD-SSIM calculation depends on interpolating polynomials based on a set of rate-distorion data points. Most implementations of BD-PSNR use exactly four rate-distortion data points for polynomial interpolation, resulting in a single number for BD-PSNR.

Advantages

BD metrics have the advantage that they are compact and in some sense more accurate representations of the quality difference compared to R-D curves alone. In case of a large number of tests, BD metrics can readily show the difference between two encoding solutions under various parameters. Further, BD metrics can consolidate results from several tests into a single chart, while showing video quality of one encoding solution with respect to another; these presentations can effectively convey an overall picture of such quality comparisons.

Limitations

The BD metrics are very useful in comparing two encoding solutions. However, for ultra-high-definition (UHD) video sequences, the BD metrics can give unexpected results.[25] The behavior appears owing to polynomial curve-fitting and the high-frequency noise in the video sequences. Standard polynomial interpolation is susceptible to Runge's phenomenon (problematic oscillation of the interpolated polynomial) when using high-degree polynomials. Even with just four data points (third degree polynomial), some interpolated curves see oscillation that can result in inaccurate BD-PSNR evaluations.

Alternative interpolation methods such as splines reduce the error caused by Runge's phenomenon and still provide curves that fit exactly through the measured rate-distortion data points. There are video examples where using piecewise cubic spline interpolation improves the accuracy of BD-PSNR calculation by nearly 1 dB over polynomial interpolation.

When oscillation occurs from polynomial interpolation, the resulting BD-PSNR calculation can be dramatically skewed. Figure 4-15 shows the polynomial interpolation problem in rate-PSNR curves from two sample encoding. The charts show the difference between polynomial interpolation and cubic spline interpolation and the BD-PSNR values using each method.

[25] Sharp Corporation, "On the Calculation of PSNR and Bit Rate Differences for the SVT Test Data," ITU SG16, Contibution 404, April 2008, available at http://www.docstoc.com/docs/101609255/On-the-calculation-of-PSNR-and-bit-rate-differences-for-the-SVT-test.

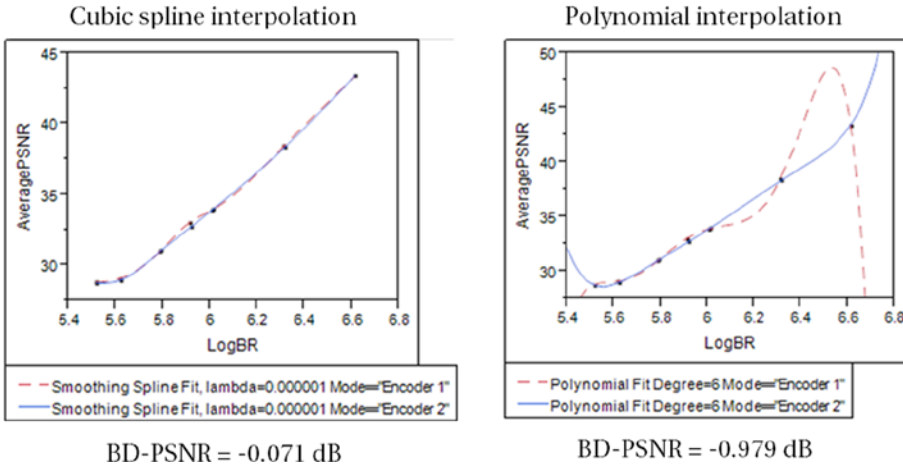


Figure 4-15. *Polynomial interpolation issue in R-D curves*

The average PSNR and bit rates correlate very closely between the two encoders, indicating that the BD-PSNR value achieved using polynomial interpolation would not be an accurate representation of the quality difference between the two encoders.

Additionally, BD-PSNR does not consider the coding complexity, which is a critical problem for practical video applications, especially for those on handheld devices whose computational capability, memory, and power supply are all limited. Such limitations are addressed by considering a generalized BD-PSNR metric that includes complexity in addition to rate and distortion. The generalized metric is presented in the next section.

Generalized BD-PSNR

The *Generalized BD-PSNR*[26] (GBD-PSNR) is a coding efficiency measure developed by generalizing BD-PSNR from R-D curve fitting to rate-complexity-distortion (R-C-D) surface fitting. GBD-PSNR involves measurement of coding complexity, R-C-D surface fitting, and the calculation of differential PSNR between two R-C-D surfaces.

In general, coding complexity is multi-dimensional and requires consideration of several factors, including the computational complexity measured by executing time or machine cycles, data cache size, memory access bandwidth, storage complexity, instruction cache size, parallelism, and pipelining. However, in practice, it is difficult to simultaneously account for all these dimensions. A widely used alternative is the coding time on a given platform. Not only does it indicate the computational complexity, but it also partially reflects the contributions from other complexity dimensions such as memory access in the coding process.

[26]X. Li, M. Wien, and J.-R. Ohm, "Rate-Complexity-Distortion Evaluation for Hybrid Video Coding," *Proceedings of IEEE International Conference on Multimedia and Expo,* (July 2010): 685–90.

In order to perform R-C-D surface fitting, the R-C-D function is defined as follows:

> **Definition 4-1.** The rate-complexity-distortion function $D(R, C)$ is the infimum of distortion D such that the rate-complexity-distortion triplet (R, C, D) is in the achievable rate-complexity-distortion region of the source for a given rate-complexity pair (R, C).

Therefore, $D(R, C)$ is non-increasing about R and C, respectively. Similar to convex R-D function, the R-C-D function is convex as well. Based on these properties, $D(R, C)$ can be approximated using an exponential model. To obtain a good tradeoff between accuracy and fitting complexity while keeping backward compatibility with BD-PSNR, $D(R, C)$ is approximated as:

$$D(R,C) = a_0 r^3 + a_1 r^2 + a_2 r + a_3 + a_4 c^2 + a_5 c \qquad \text{(Equation 4-26)}$$

where, $a_0, \ldots, a_5$ are fitting parameters, $r = \log(r)$, $c = \log(C)$, R is the output bit rate, C is the coding complexity, and D is the distortion in terms of PSNR. To fit an R-C-D surface with this equation, at least six (R, C, D) triplets from actual coding are necessary. However, in practice, a higher number of (R, C, D) triplets will lead to a better accuracy. Typically 20 data points are used to fit such a surface.

Similar to BD-PSNR, the average differential PSNR between two R-C-D surfaces can be calculated as:

$$\Delta P_{GBD} = \frac{\int_{r_L}^{r_H} \int_{c_L}^{c_H} (D_2(r,c) - D_1(r,c)) dc dr}{(r_H - r_L)(c_H - c_L)} \qquad \text{(Equation 4-27)}$$

where DP_{GBD} is the GBD-PSNR, $D_1(r, c)$ and $D_2(r, c)$ are the two fitting functions for the two R-C-D surfaces, r_H, r_L, c_H, and c_L are the logarithmic forms of R_H, R_L, C_H, and C_L which bound the overlapped R-C region from the actual coding results.

Due to the complexity nature of some algorithms, the two R-C-D surfaces may have no R-C intersection. In this extreme case, the GBD-PSNR is undefined.

Limitations

The dynamic range of coding complexity covered by GBD-PSNR is sometimes limited. This happens when the coding complexity of the two encoders are so different that there is only a relatively small overlapped region by the two R-C-D surfaces.

Also, the coding complexity is platform and implementation dependent. Although GBD-PSNR shows a good consistency over different platforms, slightly different GBD-PSNR value may still be obtained on different platforms.

Examples of Standards-based Measures

There are a few objective quality measures based on the ITU-T standards.

Video Quality Metric

The video quality metric (VQM)[27] is an objective measurement for perceived video quality developed at the National Telecommunications and Information Administration (NTIA). Owing to its excellent performance in the VQEG Phase 2 validation tests, the VQM methods were adopted by the American National Standards Institute (ANSI) as a national standard, and by ITU as ITU-T Rec. J. 144.[28] The VQM measures the perceptual effects of video impairments, including blurring, jerkiness, global noise, block distortion, and color distortion, and combines them into a single metric. The testing results show that VQM has a high correlation with subjective video quality assessment.

The algorithm takes a source video clip and a processed video clip as inputs and computes the VQM in four steps:

1. *Calibration:* In this step the sampled video is calibrated in preparation for feature extraction. The spatial and temporal shift, the contrast and the brightness offset of the processed video are estimated and corrected with respect to the source video.

2. *Quality Features Extraction*: In this step, using a mathematical function, a set of quality features that characterize perceptual changes in the spatial, temporal, and color properties are extracted from spatio-temporal subregions of video streams.

3. *Quality Parameters Calculation:* In this step, a set of quality parameters that describe perceptual changes in video quality are computed by comparing the features extracted from the processed video with those extracted from the source video.

4. *VQM Calculation***:** VQM is computed using a linear combination of parameters calculated from the previous steps.

VQM can be computed using various models based on certain optimization criteria. These models include television model, video conferencing model, general model, developer model, and PSNR model. The general model uses a linear combination of seven parameters. Four of these parameters are based on features extracted from spatial gradients of the luma component, two parameters are based on features extracted from the vector formed by the two chroma components, and the last parameter is based on contrast and absolute temporal information features, both extracted from the luma component. Test results show a high correlation coefficient of 0.95 between subjective tests and the VQM general model (VQMG).[27]

---

[27]M. Pinson, and S. Wolf, "A New Standardized Method for Objectively Measuring Video Quality," *IEEE Transactions on Broadcasting* 50, no. 3 (September 2004): 312–22.

[28]*ITU-T Recommendation J.144: Objective Perceptual Video Quality Measurement Techniques for Digital Cable Television in the Presence of a Full Reference* (Geneva, Switzerland: International Telecommunications Union, 2004).

ITU-T G.1070 and G.1070E

The ITU Recommendation G.1070[29] is a standard computational model for *quality of experience* (QoE) planning. Originally developed for two-way video communication, G.1070 model has been widely used, studied, extended, and enhanced. In G.1070, the visual quality model is based on several factors, including frame rate, bit rate, and packet-loss rate. For a fixed frame rate and a fixed packet-loss rate, a decrease in bit rate would result in a corresponding decrease in the G.1070 visual quality. However, a decrease in bit rate does not necessarily imply a decrease in quality. It is possible that the underlying video content is of low complexity and easy to encode, and thus results in a lower bit rate without corresponding quality loss. G.1070 cannot distinguish between these two cases.

Given assumptions about the coding bit rate, the frame rate, and the packet-loss rate, the G.1070 video quality estimation model can be used to generate an estimate, typically in the form of a quality score, of the perceptual quality of the video that is delivered to the end user. This score is typically higher for higher bit rates of compressed videos, and lower for lower bit rates of compressed videos.

To calculate the G.1070 visual quality estimate, a typical system includes a data collector or estimator that is used to analyze the encoded bitstream, extract useful information, and estimate the bit rate, frame rate, and packet-loss rate. From these three estimates, a G.1070 Video Quality Estimator computes the video quality estimate according to a function defined in Section 11.2 of Rec. G.1070.

Although the G.1070 model is generally suitable for estimating network-related aspects of the perceptual video quality, such as the expected packet-loss rate, information about the content of the video is generally not considered. For example, a video scene with a complex background and a high level of motion, and another scene with relatively less activity or texture, may have dramatically different perceived qualities even if they are encoded at the same bit rate and frame rate. Also, the coding bit rate required to achieve high-quality coding of an easy scene may be relatively low. Since the G.1070 model generally gives low scores for low-bit-rate videos, this model may unjustifiably penalize such easy scenes, notwithstanding the fact that the perceptual quality of that video scene may actually be high. Similarly, the G.1070 score can overestimate the perceptual quality of video scenes. Thus, the G.1070 model may not correlate well with subjective quality scores of the end users.

To address such issues, a modified G.1070 model, called the G.1070E was introduced.[30] This modified model takes frame complexity into consideration, and provides frame complexity estimation methods. Based on the frame complexity, bit-rate normalization is then performed. Finally, the G.1070 Video Quality Estimator uses the normalized bit rate along with the estimated frame rate and packet-loss rate to yield the video quality estimate.

---

[29] *ITU-T Recommendation G.1070: Opinion Model for Video-Telephony Applications* (Geneva, Switzerland: International Telecommunications Union, 2012).

[30] B. Wang, D. Zou, R. Ding, T. Liu, S. Bhagavathi, N. Narvekar, and J. Bloom, "Efficient Frame Complexity Estimation and Application to G.1070 Video Quality Monitoring," *Proceedings of 2011 Third International Workshop on Quality of Multimedia Experience* (2011): 96–101.

The G.1070E is a no-reference compressed domain objective video-quality measurement model. Experimental results show that the G.1070E model yields a higher correlation with subjective MOS scores and can reflect the quality of video experience much better than G.1070.

ITU-T P.1202.2

The ITU-T P.1202 series of documents specifies models for monitoring the video quality of IP-based video services based on packet-header and bitstream information. Recommendation ITU-T P.1202.2[31] specifies the algorithmic model for the higher-resolution application area of ITU-T P.1202. Its applications include the monitoring of performance and quality of experience (QoE) of video services such as IPTV. The Rec. P.1202.2 and has two modes: Mode 1, where the video bitstreams are parsed and not decoded into pixels, and Mode 2, where the video bitstreams are fully decoded into pixels for analyzing.

The Rec. P.1202.2 is a no-reference video-quality metric. An implementation of the algorithm has the following steps:

1. Extraction of basic parameters such as frame resolution, frame level quantization parameter, frame size, and frame number.
2. Aggregation of basic parameters into internal picture level to determine frame complexity.
3. Aggregation of basic parameters into model level to obtain video sequence complexity, and quantization parameter at the video sequence level.
4. Quality estimation model to estimate the MOS as:

$$P.1202.2\ MOS = f(frame\ QP, frame\ resolution, frame\ size, frame\ number) \quad \text{(Equation 4-28)}$$

Studies have found that the P.1202.2 algorithm's estimated MOS has similar Pearson linear correlation coefficient and Spearman ranked order correlation coefficient to VQEG JEG's (Joint Effort Group) estimated MOS, which uses the following linear relationship:[32]

$$VQEG\ JEG\ MOS = -0.172 \times frame\ QP + 9.249 \quad \text{(Equation 4-29)}$$

However, both of these results are worse than MS-SSIM. It is also found that P.1202.2 does not capture compression artifacts well.

---

[31] *ITU-T Recommendation P.1202.2: Parametric Non-intrusive Bitstream Assessment of Video Media Streaming Quality – Higher Resolution Application Area* (Geneva, Switzerland: International Telecommunications Union, 2013).

[32] L. K. Choi, Y. Liao, B. O'Mahony, J. R. Foerster, and A. C. Bovik, "Extending the Validity Scope of ITU-T P.1202.2," in *Proceedings of the 8th International Workshop on Video Processing and Quality Metrics for Consumer Electronics* (Chandler, AZ: VPQM, 2014), retrieved from www.vpqm.org.

Therefore, an improved FR MOS estimator is proposed based on MS-SSIM. In particular, an MS-SSIM-based remapping function is developed. The resulting estimated MOS is a function of MS-SSIM and the frame parameters, such as frame level quantization parameter, frame size, frame type, and resolution. The algorithm first performs devices and content analysis, followed by spatial complexity computation.

Then, a non-linear model fitting is performed using logistic function. These results, along with the MS-SSIM values, are provided to the MOS estimator to calculate the estimated MOS. Experimental results show that for a set of tests, the estimated MOS has a Pearson correlation coefficient >0.9 with MOS, which is much better than that given by MS-SSIM (0.7265).

Measurement of Video Quality

We elaborate on important considerations for video quality measurement, for both subjective and objective measurements. Further, for clarity we discuss the objective measurements from typical application point of view.

Subjective Measurements

The metrics used in subjective measurement are MOS and DMOS. However, after obtaining the raw scores, they cannot be directly used. To eliminate bias, the following measurement procedure is generally used.

Let s_{ijk} denote the score assigned by subject i to video j in session k. Usually, two sessions are held. In the processing of the raw scores, difference scores d_{ijk} are computed per session by subtracting the quality assigned by the subject to a video from the quality assigned by the same subject to the corresponding reference video in the same session. Computation of difference scores per session helps account for any variability in the use of the quality scale by the subject between sessions. The difference scores are given as:

$$d_{ijk} = s_{ijk} - s_{ijrefk}. \quad \text{(Equation 4-30)}$$

The difference scores for the reference videos are 0 in both sessions and are removed. The difference scores are then converted to Z-scores per session:

$$\mu_{ik} = \frac{1}{N_{ik}} \sum_{j=1}^{N_{ik}} d_{ijk} \quad \text{(Equation 4-31)}$$

$$\sigma_{ik} = \sqrt{\frac{1}{N_{ik} - 1} \sum_{j=1}^{N_{ik}} (d_{ijk} - \mu_{ik})^2} \quad \text{(Equation 4-32)}$$

$$z_{ijk} = d_{ijk} - \mu_{ik}\sigma_{ik} \quad \text{(Equation 4-33)}$$

where N_{ik} is the number of test videos seen by subject i in session k.

Every subject sees each test video in the database exactly once, either in the first session or in the second session. The Z-scores from both sessions are then combined to create a matrix $\{z_{ij}\}$. Scores from unreliable subjects are discarded using the procedure specified in the ITU-R BT.500-13 recommendation.

The distribution of the scores is then investigated. If the scores are normally distributed, the procedure rejects a subject whenever more than 5 percent of scores assigned by that subject fall outside the range of two standard deviations from the mean scores. If the scores are not normally distributed, the subject is rejected whenever more than 5 percent of his scores fall outside the range of 4.47 standard deviations from the mean scores. However, in both situations, subjects who are consistently pessimistic or optimistic in their quality judgments are not eliminated.

The Z-scores are then linearly rescaled to lie in the range [0,100]. Finally, the DMOS of each video is computed as the mean of the rescaled Z-scores from the remaining subjects after subject rejection.

Objective Measurements and Their Applications

Objective measurements are very useful in automated environments—for example, in automated quality comparison of two video encoder solutions. Figure 4-16 shows the block diagram of a typical encoder comparison setup using full-reference objective video-quality metrics.

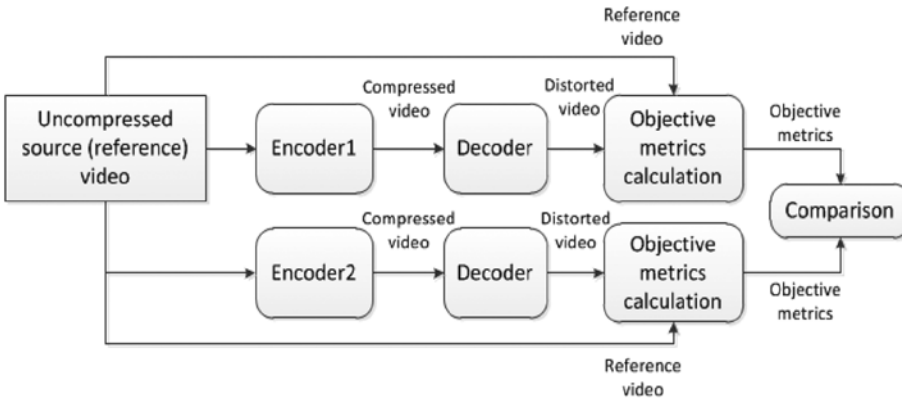


Figure 4-16. *An example of a typical encoder comparison setup using FR objective quality metrics*

Several factors need to be considered for such an application using full reference objective video quality metrics:

- The source and distorted videos need to be aligned in time so that the same video frame is compared for quality.
- The same decoder implementation should be used, eliminating any measurement variability owing to the decoding process.

- To ensure a fair comparison, the encoder parameters must be the same or as close as possible.
- No pre-processing is assumed before the encoding process. Although it is possible to use a pre-processing step before each encoder, in that case the same pre-processor must be used.

Notice that such a setup can take advantage of automation and use an enormous set of video clips for comparison of different encoder implementations, thus exposing the strengths and weaknesses of each encoder under various workload complexities. Such source comparisons without considering network or channel errors are ideal for a fair comparison. However, in practical applications, such as mobile video recording using two different devices, where the recorded videos are stored and decoded before computing objective quality metrics, quality comparison should be done in similar environments as much as possible. For example, in wireless network environment, the packet-loss rate or bit-error rate should be similar.

Objective measures are also extensively used to determine frame drops in video applications. For example, as the (distorted) video is consumed, frame drops can be detected if the PSNR between the source and the distorted video is tracked frame by frame. In low-distortion environments, the consumed video would reasonably match the source; so the PSNR would also be a typical number (e.g., 25–40 dB) depending on the lossy characteristics of the various channels that introduce errors. However, in case of a frame drop, the wrong frame would be compared against the source, and a very low PSNR would be obtained, indicating the frame drop. This effect is exaggerated when the video contains frequent scene changes.

The same concept can be applied to detect sudden low-quality frames with corruption or other artifacts in a video. Such corruption can happen owing to network errors or encoder issues. But a sudden drop in PSNR or other objective measures can indicate the location of the corruption in the video in an automated environment.

Parameters to Tune

In visual communication applications, video codecs are one of the main sources of distortions. Since video decoders must follow certain specifications as defined by various standards, decoders generally do not significantly contribute to video-quality degradation. However, encoders are free to design and implement algorithms to control the amount of compression and thereby the amount of information loss depending on various considerations for system resources, application requirements, and the application environment. Therefore, in the video-encoding applications, there are several parameters that dictate the amount of information loss and thus influence the final video quality. Some of these parameters are adjustable at the algorithm level by the system architects; some are tunable by the implementors, while few parameters are usually available to the users for tuning.

Parameters that Impact Video Quality

It is very important to understand the impact of the following parameters on final visual quality, particularly for benchmarking, optimization, or comparative analysis of the video encoding solutions.

- **Number of lines in the vertical display resolution:** High-definition television (HDTV) resolution is 1,080 or 720 lines. In contrast, standard-definition digital television (DTV) is 480 lines (for NTSC, where 480 out of 525 scanlines are visible) or 576 lines (for PAL/SECAM, where 576 out of 625 scanlines are visible). For example, the so-called *DVD quality* is standard definition, while Blu-ray discs are high definition. An encoder may choose to reduce the resolution of the video as needed, depending on the available number of bits and the target quality level. However, recent encoders typically process the full resolution of video in most applications.

- **Scanning type:** Digital video uses two types of image scanning pattern: progressive scanning or interlaced scanning. Progressive scanning redraws all the lines of a video frame when refreshing the frame, and is usually denoted as 720p or 1080p, for example. Interlaced scanning draws a *field*—that is, every other line of the frame at a time—so the *odd numbered* lines are drawn during the first refresh operation and then the remaining *even numbered* lines are drawn during a second refreshing. Thus, the interlaced refresh rate is double that of the progressive referesh rate. Interlaced scanned video is usually denoted as 480i or 1080i, for example.

  Movement of object makes a difference in perceived quality of interlaced scanned video. On a progressively scanned display, interlaced video yields better quality for still objects in frames owing to the higher refresh rate, but loses up to half of the resolution and suffers *combing* artifacts when objects in a frame is moving. Note that combing artifacts only occur when two fields are woven together to form a single frame and then displayed on a progressive display. Combing artifacts do not occur when interlaced content is shown on an interlaced display and when different deinterlacing algorithms such as *bob* are used for display on progressive monitors.

  In practice, two interlaced fields formulate a single frame because the two fields consisting of the odd and even lines of one frame are temporally shifted. Frame pulldown and segmented frames are special techniques that allow transmitting full frames by means of interlaced video stream. For appropriate reconstruction and presentation at the receiving end of a transmission system, it is necessary to track whether the top or bottom field is transmitted first.

- **Number of frames or fields per second (Hz):** In Europe, 50 Hz television broadcasting system is more common, while in the United States, it is 60 Hz. The well-known 720p60 format is 1280×720 pixels, progressive encoding with 60 frames per second (60 Hz). The 1080i50/1080i60 format is 1920×1080 pixels, interlaced encoding with 50/60 fields, (50/60 Hz) per second. If the frame/field rate is not properly maintained, there may be visible *flickering* artifact. Frame drop and frame jitter are typical annoying video-quality issues resulting from frame-rate mismanagement.

- **Bit rate:** The amount of compression in digital video can be controlled by allocating a certain number of bits for each second's worth of video. The bit rate is the primary defining factor of video quality. Higher bit rate typically implies higher quality video. Efficient bit allocation can be done by taking advantage of skippable macroblocks and is based on the spatio-temporal complexity of macroblocks. The amount of quantization is also determined by the available bit rate, thereby highly impacting the *blocking* artifact at transform block boundaries.

- **Bit-rate control type:** The bit-rate control depends on certain restrictions of the transmission system and the nature of the application. Some transmission systems have fixed channel bandwidth and need video contents to be delivered at a constant bit rate (CBR), while others allow a variable bit rate (VBR), where the amount of data may vary per time segment. CBR means the decoding rate of the video is constant. Usually a decoding buffer is used to keep the decoded bits until a frame's worth of data is consumed instantaneously. CBR is useful in streaming video applications where, in order to meet the requirement of fixed number of bits per second, stuffing bits without useful information may need to be transmitted.

  VBR allows more bits to be allocated for the more complex sections of the video, and fewer bits for the less complex sections. The user specifies a given subjective quality value, and the encoder allocates bits as needed to achieve the given level of quality. Thus a more perceptually consistent viewing experience can be obtained using VBR. However, the resulting compressed video still needs to fit into the available channel bandwidth, necessitating a maximum bit rate limit. Thus, the VBR encoding method typically allows the user to specify a bit-rate range indicating a maximum and/or minimum allowed bit rate. For storage applcations, VBR is typically more appropriate compared to CBR.

  In addition to CBR and VBR, the average bit rate (ABR) encoding may be used to ensure the output video stream achieves a predictable long-term average bit rate.

- **Buffer size and latency:** As mentioned above, the decoding buffer temporarily stores the received video as incoming bits that may arrive at a constant or variable bit rate. The buffer is drained at specific time instants, when one frame's worth of bits are taken out of the buffer for display. The number of bits that are removed is variable depending on the frame type (intra or predicted frame). Given that the buffer has a fixed size, the bit arrival rate and the drain rate must be carefully maintained such that the buffer does not overflow or be starved of bits. This is typically done by the rate control mechanism that governs the amount of quantization and manages the resulting frame sizes. If the buffer overflows, the bits will be lost and one or more frames cannot be displayed, depending on the frame dependency. If it underflows, the decoder would not have data to decode, the display would continue to show the previously displayed frame, and decoder must wait until the arrival of a decoder refresh signal before the situation can be corrected. There is an initial delay between the time when the buffer starts to fill and the time when the first frame is taken out of the buffer. This delay translates to the decoding latency. Usually the buffer is allowed to fill at a level between 50 and 90 percent of the buffer size before the draining starts.

- **Group of pictures structure:** The sequence of dependency of the frames is determined by the frame prediction structure. Recall from Chapter 2 that intra frames are independently coded, and are usually allocated more bits as they typically serve as anchor frames for a group of pictures. Predicted and bi-predicted frames are usually more heavily quantized, resulting in higher compression at the expense of comparatively poor individual picture quality. Therefore, the arrangement of the group of picture is very important. In typical broadcast applications, intra frames are transmitted twice per second. In between two intra frames, the predicted and bi-predicted frames are used so that two bi-predicted frames are between the predicted or intra reference frames. Using more bi-predicted frames does not typically improve visual quality, but such usage depends on applications. Note that, in videos with rapidly changing scenes, predictions with long-term references are not very effective. Efficient encoders may perform scene analysis before determining the final group of pictures structure.

- **Prediction block size:** Intra or inter prediction may be performed using various block sizes, typically from 16×16 down to 4×4. For efficient coding, suitable sizes must be chosen based on the pattern of details in a video frame. For example, an area with finer details can benefit from smaller prediction block sizes, while a flat region may use larger prediction block sizes.

- **Motion parameters:** Motion estimation search type, search area, and cost function play important roles in determining visual quality. A full search algorithm inspects every search location to find the best matching block, but at the expense of very high computational complexity. Studies have suggested that over 50 percent of the encoding computations are spent in the block-matching process. The number of computations also grows exponentially as the search area becomes larger to capture large motions or to accommodate high-resolution video. Further, the matching criteria can be selected from techniques such as sum of absolute difference (SAD) and sum of absolute transformed differences (SATD). Using SATD as the matching criteria provides better video quality at the expense of higher computational complexity.
- **Number of reference pictures:** For motion estimation, one or more reference pictures can be used from lists of forward or backward references. Multiple reference pictures increase the probability of finding a better match, so that the difference signal is smaller and can be coded more efficiently. Therefore, the eventual quality would be better for the same overall number of bits for the video. Also, depending on the video content, a frame may have a better match with a frame that is not an immediate or close neighbor. This calls for long-term references.
- **Motion vector precision and rounding:** Motion compensation can be performed at various precision levels: full-pel, half-pel, quarter-pel, and so on. The higher the precision, the better the probability of finding the best match. More accurate matching results in using fewer bits for coding the error signal, or equivalently, using a finer quantization step for the same number of bits. Thus quarter-pel motion compensation provides better visual quality for the same number of bits compared to full-pel motion compensation. The direction and amount of rounding are also important to keep sufficient details of data, leading to achieving a better quality. Rounding parameters usually differ based on intra or inter type of prediction blocks.
- **Interpolation method for motion vectors:** Motion vector interpolation can be done using different types of filters. Typical interpolation methods employ a bilinear, a 4-tap, or a 6-tap filter. These filters produce different quality of the motion vectors, which leads to differences in final visual quality. The 6-tap filters generally produce the best quality, but are more expensive in terms of processing cycles and power consumption.

- **Number of encoding passes:** Single-pass encoding analyzes and encodes the data *on the fly*. It is used when the encoding speed is most important—for example, in real-time encoding applications. Multi-pass encoding is used when the encoding quality is most important. Multi-pass encoding, typically implemented in two passes, takes longer than single-pass, as the input data goes through additional processing in each pass. In multi-pass encoding, one or more initial passes are used to collect the video characteristics data, and a final pass uses that data to achieve uniform quality at a specified target bit rate.
- **Entropy coding type:** Entropy coding type such as CABAC or CAVLC does not generally impact video quality. However, if there is a bit-rate limit, owing to the higher coding efficiency, CABAC may yield better visual quality, especially for low-target bit rates.

Tradeoff Opportunities

Video encoders usually have tunable parameters to achieve the best possible quality or the best possible encoding speed for that encoder. Some parameters allow the encoder to analyze the input video and collect detailed information of the characteristics of the input video. Based on this information, the encoder makes certain decisions regarding the amount of compression to perform or the encoding mode to be used. Often, multiple passes are used for the analysis and subsequent encoding. Thus, the encoder is able to compress the video efficiently and achieve the best possible quality for the given algorithm. However, such analysis would require time and would slow down the encoding process. Further, the analysis work would increase the power consumption of the encoding device. Therefore, sometimes tuning of the certain parameters to adapt to the given video characteristics is not attempted in order to increase performance, or to meet system resource constraints. Rather, these parameters use pre-defined values for this purpose, thereby reducing analysis work and aiming to achieve the best possible speed.

Most of the parameters mentioned in the above section that affect visual quality also affect the encoding speed. To achieve a good tradeoff between quality and speed for a given video encoder, several parameters can be tuned. Although not all parameters listed here are tunable by the end user, depending on the encoder implementation, some parameters may be exposed to the end-user level.

- **Bit rate, frame rate, resolution:** Videos with high bit rate, frame rate, and resolution usually take longer to encode, but they provide better visual quality. These parameters should be carefully set to accommodate the application requirement. For example, real-time requirements for encode and processing may be met on a certain device with only certain parameters.
- **Motion estimation algorithm:** There are a large number of fast-motion estimation algorithms available in the literature, all of which are developed with a common goal: to increase the speed of motion estimation while providing reasonable quality. Since motion estimation is the most time-consuming part of video encoding, it is very important to choose the algorithm carefully.

- **Motion search range:** For best quality, the motion search range should be set to a high value so that large motions can be captured. On the other hand, the larger the search window, the more expensive is the search in terms of amount of computation to be done. So, a large search area directly impacts the encoding speed, memory bandwidth, frame latency, and power consumption. In addition, the large motion vectors would require more bits to encode. If the difference signal between a source block and the predicted block has substantial energy, it may be worthwhile to encode the block in intra-block mode instead of using the large motion vectors. Therefore, a tradeoff needs to be made between the search parameters and coding efficiency in terms of number of bits spent per decibel of quality gain.
- **Adaptive search:** To achieve better quality, often the motion search algorithms can adapt to the motion characteristics of the video and can efficiently curb the search process to gain significant encoding speed. For example, in order to accelerate motion search, an algorithm can avoid searching the stationary regions, use switchable shape search patterns, and take advantage of correlations in motion vectors. Thus, encoding speed can be increased without resorting to suboptimal search and without sacrificing visual quality.
- **Prediction types:** Predicted and bi-predicted frames introduce various levels of computational complexity and generally introduce visual quality loss in order to achieve compression. However, they also provide visually pleasing appearance of smooth motion. Therefore, prediction type of a frame is an important consideration in tradeoffs between quality and encoding speed.
- **Number of reference frames:** Mutiple reference frames can provide better visual quality than single reference frames, but computing motion vectors from multiple references are more time-consuming. In resource constrained environment, such parameters are important factors in tradeoff considerations.
- **Transform mode and partition size:** A block may use 8×8 or 4×4 sizes for the transform and various partition sizes for the prediction. On some platforms, processing four 4×4 blocks may be slower than processing one 8×8 block. However, depending on the amount of details available in the video, such decision may impact the visual quality, as 4×4 partitions have better adaptability to finer details.

- **Skip conditions:** A block can be skipped if it meets certain criteria. Better skip decisions can be made based on analysis of the quantized transform coefficients characteristics compared to simple heuristics, resulting in better quality. But a large amount of computation is necessary to adopt such complex algorithms. It is a clear tradeoff opportunity for resource-constrained devices.
- **Deblocking filter parameters:** Encoding speed is usually sensitive to deblocking filter parameters. Performing strong deblocking slows the encoding, but depending on the content and the amount of blocking artifact, it may provide significantly better visual quality.

Summary

This chapter discussed visual quality issues and factors impacting the perceptual quality of video to a human observer. First, we studied the various compression and processing artifacts that contribute to visual quality degradation, and various factors that affect visual quality in general. Next, we discussed various subjective and objective quality evaluation methods and metrics with particular attention to various ITU-T standards. We discussed several objective quality evaluation approaches in detail. These approaches are based on various factors: error-sensitivity, structural similarity, information fidelity, spatio-temporal, saliency, network awareness, and noise. We also discussed video coding efficiency evaluation metrics and some examples of standard-based algorithms.

In the final part of this chapter, we covered about the encoding parameters that primarily impact video quality. Tuning some parameters offer good tradeoff opportunities beween video quality and compression speed. These include bit rate, frame rate, resolution, motion estimation parameters, Group of Pictures structure, number of reference frames, and deblocking filter parameters. Some of these parameters may be available to the end user for tuning.