Low-Complexity Video Watermarking Using Mean Block Luminance Quantization

Sergey Putilin, Dmitriy Kulikov
Department of Computational Mathematics and Cybernetics
Moscow State University, Moscow, Russia
{sputilin, dkulikov}@graphics.cs.msu.ru

Internet multimedia services giving opportunity to share videos are growing rapidly nowadays. Together with new video editing tools and peer-to-peer networks, these services make the problem of protecting video intellectual property very important. Big progress in this area is already made, but it is still not sufficient.

In this paper, we propose new video watermarking algorithm based on blocks mean luminance. It is designed to extract watermark even from relatively short video sequences. The algorithm provides different robustness / visibility tradeoffs. Its robustness is tested using different distortions, such as brightness changes, contrast changes, compression with MPEG-4 codec using various bitrates, and video resizing.

The algorithm is designed to store a bit sequence or a word or a short phrase as a watermark message in a video sequence. It can be divided into two sections: a low level section and a high level section. A low level section of the algorithm operates with bits. It is in charge of storing the bits into video during watermark embedding and reading the bits during watermark decoding. A high level section operates with a watermark message itself. It is decomposing watermark message into bit sequence during watermarking the video and composing bit sequence into the message during watermark extraction.

The main idea of a low level section of algorithm is quantization of blocks mean luminance into values that correspond to 0 or 1 state of watermark information bits. During the embedding, pixels of the current image block are randomly adjusted so that the mean luminance of the block corresponds to the nearest level of the information bit. Also the adjustment algorithm is made so that the frame difference is minimized.

Two ways of improving low level algorithm robustness are used. The first one is luminance quantization step extension, which leads to increased watermark visibility. The second way is duplicating the blocks containing the bit. Different patterns of block locations can be used. This way of robustness improvement decreases the information capacity of watermarking.

A high level section of the algorithm is responsible for decomposing a watermarking message into bits and synchronization. The main idea is repeating the bits of watermark message through the entire video sequence.

Let us consider we are trying to embed a watermark message consisting of 3 bytes: byte0, byte1, byte2. According to the bit embedding algorithm described above the number of bits stored in one video frame can be calculated. The algorithm is designed to have the setting of maximum watermark message length. This setting should be the same during watermarking and extracting watermark from the video sequence.

As the current watermark message can be shorter than the maximum watermark message length, two bytes containing a current message length are added to the watermark message. After this procedure actually embedded watermark data will be like this: lenbyte0, lenbyte1, byte0, byte1, byte2, where lenbyte0 and lenbyte1 are the bytes that contain number 3 – the current watermark message length in bytes. So, we get a 5 bytes long watermark data.

In every frame, a bit sequence begins with a synchronization byte. In this byte, the number of the following bytes from watermarking data sequence is stored. The main idea is that this bytes are periodic and their sequence can be recreated during watermark extraction. After that Hamming distances are calculated between different shifts of recreated synchronization bytes sequence and synchronization bytes sequence obtained from video. The least Hamming distance corresponds to the correct shift.

After the shift is performed, watermarking data can be analyzed and decoded into original watermark message.

During evaluation the algorithm has been set up to insert one bit of watermark into 4 blocks of a video frame. Four algorithm presets were tested. An 8-byte long watermark message was used. Maximum watermark message length was set to 16 bytes. 32x32 blocks were used.

5 4CIF sequences (720x576 resolution) and two HDTV sequences (1920x1080 and 1280x720 resolution) were used for tests. All the sequences were 12 frames long. The results are shown on Figure 1 for HDTV sequences.

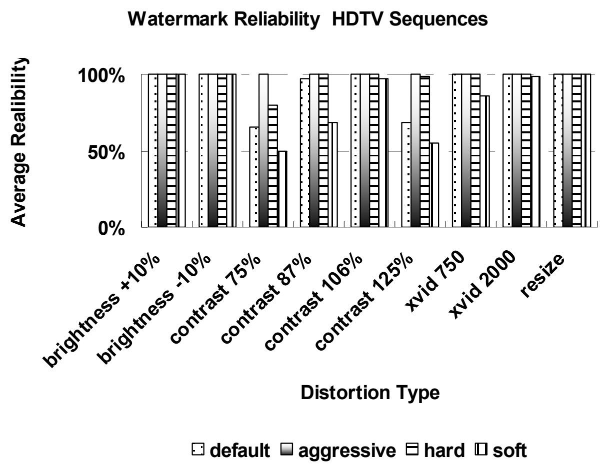

Figure 1. Watermark reliability for HDTV video sequences.

Processing speed of the algorithm is quite fast: 20 fps on 4CIF sequences on a 2GHz Core 2 Duo PC using C++ AVISynth (http://avisynth.org) plugin implementation of the algorithm.

Authors of the article express their gratitude to Dr. Dmitriy Vatolin for algorithm ideas and to Oleg Petrov for initial algorithm implementation.