CSE 126/228F Lecture 7: April 22

MPEG

Interframe Coding

With 30 frames per second, you will naturally expect differences between successive frames of a video sequence to be relatively small. MPEG achieves a great deal of compression by exploiting the relationship between successive frames. Rather than encoding one initial frame and then sending only differences for all the remaining frames, MPEG uses the *windowing* approach. Windowing breaks up the video sequence into smaller subsequences and encodes differences only within a window, not between them. This is done for two reasons,

1. **Protection from errors:** what if you lose a frame in transmission? It is possible that the entire remainder of the sequence could be useless without windowing.
2. **Random access and Editing:** How do you edit a compressed video sequence without having to decompress and then re-encode it without windowing?

Each of these windows in MPEG is called a *Group of Pictures (GOP)*. How long is a GOP? The answer: as long as you like. The length of a GOP is not specified by the MPEG standard, and a video sequence can even contain GOPs of different lengths.

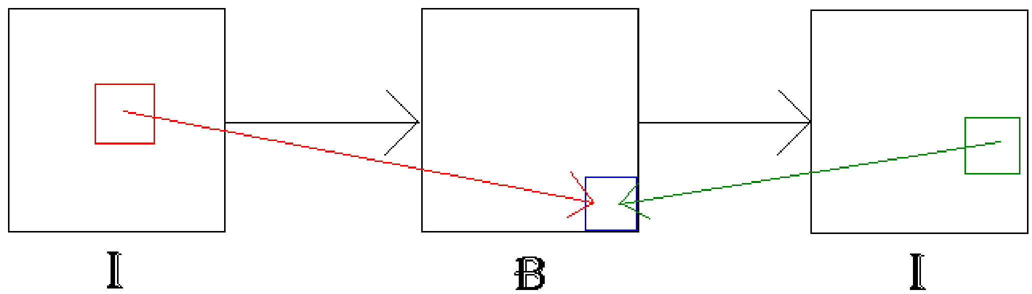
Frame Types

- **I** frames are intracoded frames. They don't depend on any other frames, you can think of them as JPEG images.
- **P** frames are predicted frames. They depend only on the previous I or P frame.
- **B** frames are bidirectional frames. They can depend on both the previous and the next I or P frames.

Because I and P frames are used to predict other P and B frames, they are called *Reference Frames*.

Motion Estimation

Motion estimation in MPEG operates on *Macroblocks*. A macroblock is a 16x16 pixel range in a frame. There are two primary types of motion estimation, forward and backward. *Forward prediction* predicts how a macroblock from the previous reference frame moves forward into the current frame. *Backward prediction* predicts how a macroblock from the next reference frame moves back into the current frame. Examples are shown below, with Forward prediction in red (left-to-right) and Backward prediction in green (right-to-left)



Motion estimation operates as follows: First, compare a macroblock of the current frame against all 16x16 regions of the frame you are predicting from. Then, select the 16x16 region with the least mean-squared error from the current macroblock and encode a *motion vector*, which specifies the 16x16 region you are predicting from and the error values for each pixel in the macroblock. This is done only for the combined Y,U, and V values. Subsampling and separation of the Y, U, and V bands comes later.

There are four types of macroblocks:

- **Forward Predicted**: (P and B only) predict from a 16x16 region in the previous reference frame
- **Backward Predicted**: (B only) predict from a 16x16 region in the next reference frame
- **Bidirectional Predicted**: (B only) predict from the average of a 16x16 region in the previous reference frame and a 16x16 region in the next reference frame.
- **Intracoded**: (I, P, or B) are not predicted, the actual pixel values are used for the macroblock.

It is important to remember that P and B frames can contain intracoded macroblocks as well as predicted macroblocks if there is no efficient way to predict the macroblock.

Decoding vs. Presentation order

MPEG is actually used in decoding order rather than presentation order. Examples of both follow:

Presentation Order

I_1 B_2 B_3 B_4 P_5 B_6 B_7 B_8 P_9 B_{10} B_{11} B_{12} I_{13}

Decoding Order

I_1 P_5 B_2 B_3 B_4 P_9 B_6 B_7 B_8 I_{13} B_{10} B_{11} B_{12}

The reason for the difference is that in order to decode a predicted frame, all frames that it may be predicting from must be decoded first. Therefore, since $B_{2..4}$ may all be predicting from both I_1 and P_5, both must be decoded before $B_{2..4}$. This distinction becomes very important when you work with MPEG.

Independent vs. Dependent GOPs

Independent GOPs do not depend on any frames of the previous GOP for prediction. Dependent GOPs depend on a reference frame from another GOP for prediction. Examples follow (in decoding order):

Case 1: GOP_1 is dependent upon GOP_2 (which starts with I_{13})

I_1 P_5 B_2 B_3 B_4 P_9 B_5 B_6 B_7 I_{13} B_{10} B_{11} B_{12}

Case 2: GOP_1 is not dependent upon GOP_2 (which starts with I_{13})

I_1 P_5 B_2 B_3 B_4 P_9 B_5 B_6 B_7 P_{12} B_{10} B_{11} I_{13}

To illustrate the difference, imagine trying to perform a simple edit operation and cut out GOP_2. In order to do this, I_{13} must be removed. If this happens, $B_{10..12}$ will not be able to be decoded since they depend on I_{13}. In the second case no frames in the first GOP depend on the second GOP, making this operation possible. As shown here, if you want to make a dependent GOP independent, end the current GOP with a P frame.

Bandwidth

Bandwidth is a major concern in digital video. For MPEG, you can remember the following:

- I frames require the most space, and give the least compression
- B frames require the least space, and give the most compression.
- P frames are in between.

MPEG-1 is fixed to a maximum 1.2Mbit/second bandwidth. If an encoded MPEG-1 stream is larger than this, the encoder will have to make the quantization more coarse (to increase compression) and re-encode the sequence. This idea is called *feedback*, where the output of the encoder is analyzed and changes the input back into the encoder until the sequence is acceptable.