UNIVERSITÀ DEGLI STUDI
DI MODENA E REGGIO EMILIA

Lecture notes for Multimedia Data Processing

Video Compression

Last updated on: 28/04/2022

What is Video?

- From a technical point of view, the video is a sequence of images, obtained by temporal sampling.
- The PAL standard, used in Europe, plans to send 25 images per second.
- A possible solution for encoding the video could be to store all the images in a single data stream containing the raw data.
- This would require a memory occupation equal to w * h * bpp * fps, where w is the width of the image, h the height, bpp the number of bits per pixel and fps (frames per second) the number of images per second.
- A 1.5-hour gray level video (8bpp) with 720 × 576 resolution at 25 fps would occupy 55.987.200.000 Bytes, that is 52.14 GB.
- Obviously, this is unacceptable, despite the technological improvements of the past few years.
- It is therefore essential to use some data compression technique.

INTRA Frame Encoding

- As with the images, it is possible to take advantage of the spatial redundancy within each frame to reduce occupancy.
- For example, the JPEG encoding, so effective for images, could be applied to every frame of the sequence.
- As for the images, this compression would lead indicatively to a compression of about 1:10, with an evident advantage, at the expense of a non-dramatic quality loss.
- This type of encoding is known as M-JPEG: Motion JPEG.
- Unfortunately, even in this way the space requirements would be too high: in the previous example, we would obtain that a 1.5-hour film would occupy (without considering the audio) 5.21 GB, approaching the maximum capacity of a DVD.
- This prompted the research to also consider the temporal aspect, or to try to use the similarities between temporally close frames.

INTER Frame Encoding

- The first possible approach is to use the previous frame to predict the current one. For example, consider two successive frames such as those shown below.
- These frames were obtained by a camera placed on a table that captures 25 frames per second.

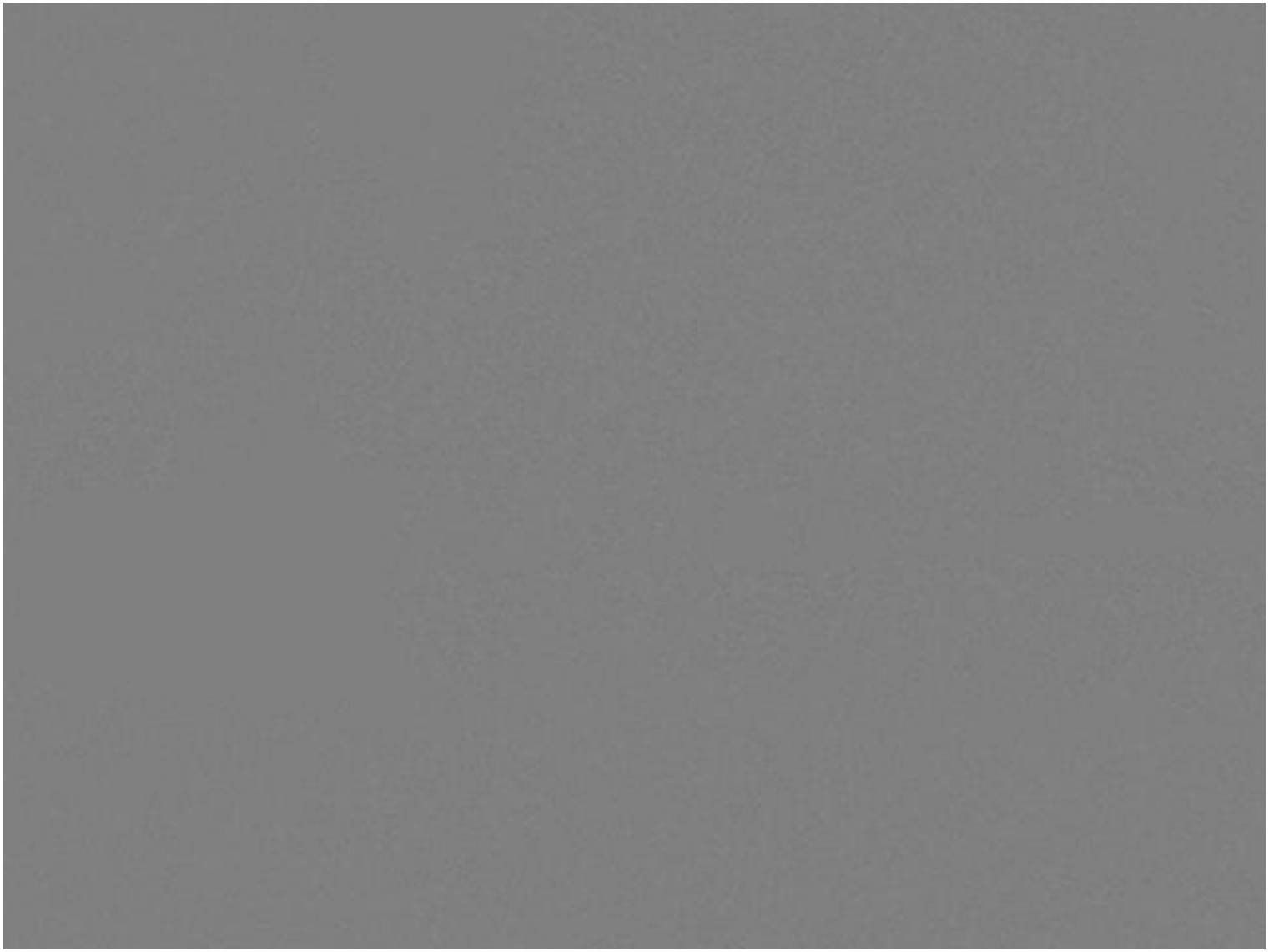
Frame A

Frame B

Time Difference

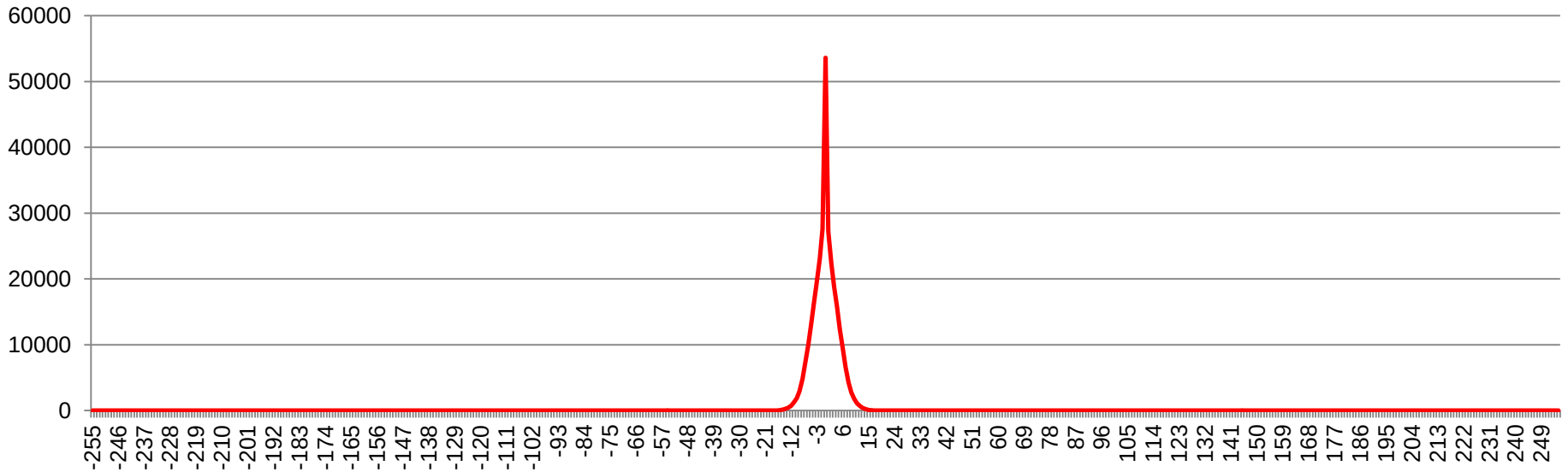
- The two frames are extremely similar. Unless some small variation, due to the acquisition, they are substantially indistinguishable.
- Assuming that it is possible to know the value of a pixel by looking at the value it had in the previous frame, we can make a prediction, by means of the difference between the pixel at the coordinates (x, y) in frame B and that in frame A.
- To visualize this difference, it is possible to create a difference image with the sole purpose of qualitatively "seeing" the effect of this operation.
- Obviously, the differences between two 8 bpp images (values in the range [0,255]) are in the range [-255,255], so to bring them back into the visible range you can divide (integer division) by 2 and add 127.
- In this way the zero will be at the value 127, the negative values will be < 127 and the positive ones > 127.

Difference Image



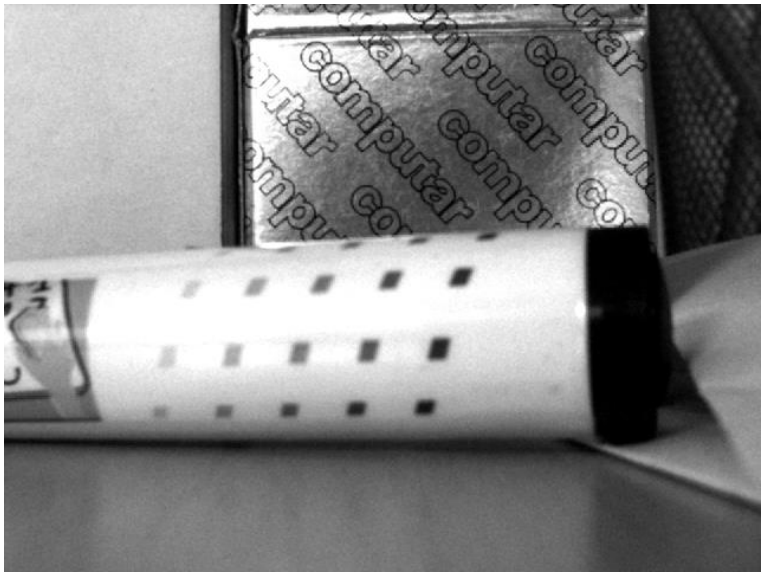
Advantages

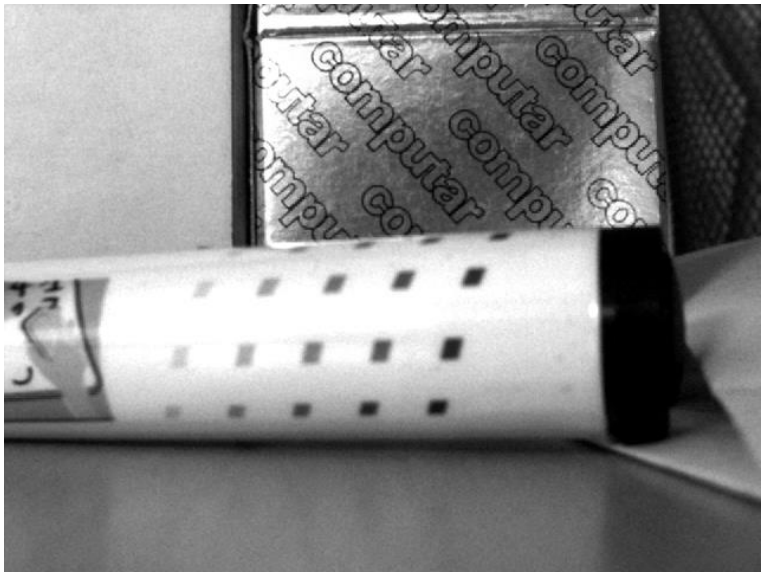
- Why is the difference advantageous?
- By observing the distribution of the difference values, it can be observed that this is very narrow and very non ununiform.
- The entropy of this distribution is 4.06, therefore using a variable length coding it is possible to halve the size of the original frame, even without loss.



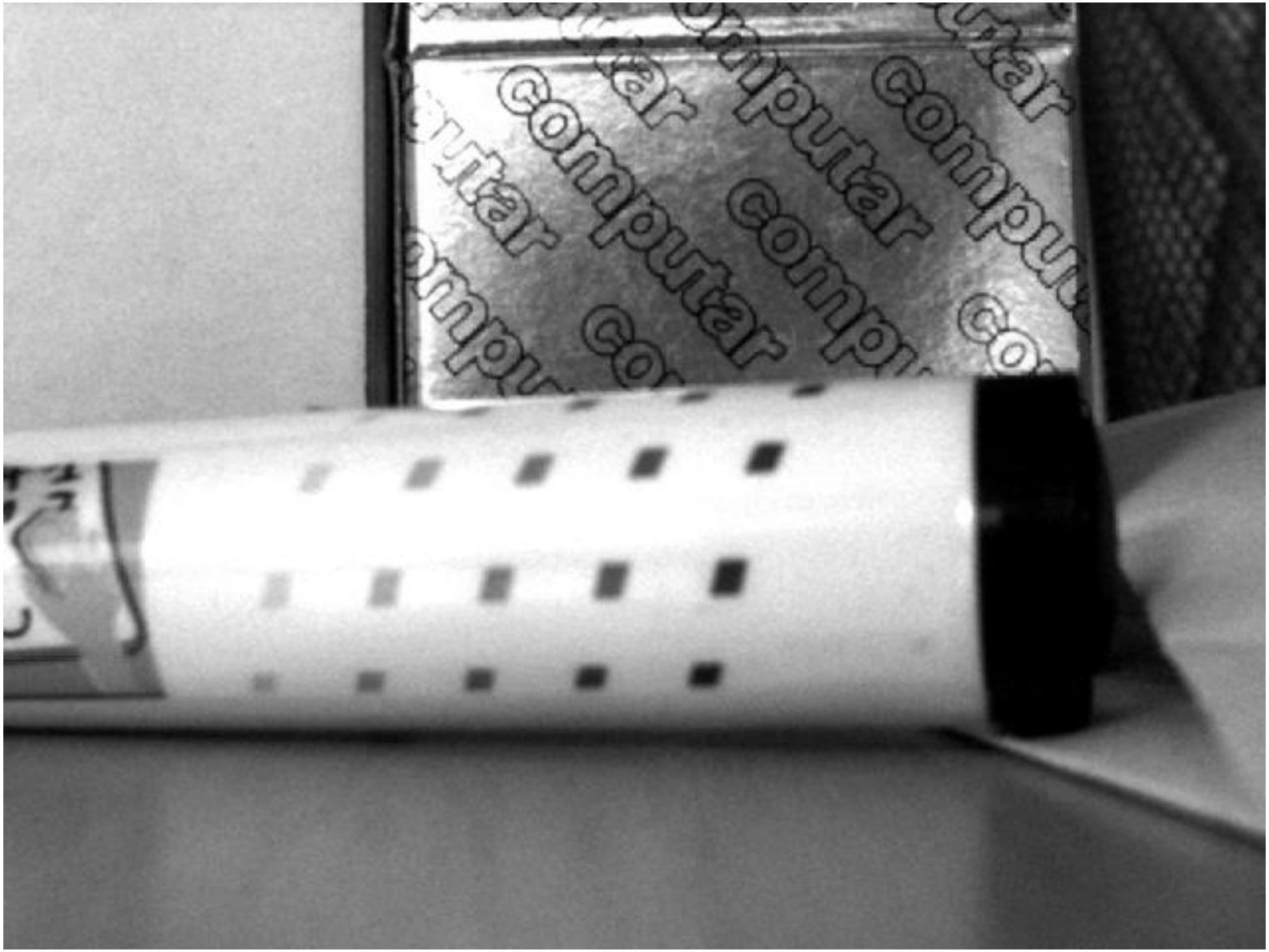
The Movement

- What has been said, however, is based on the assumption that the scene changes "little". If we introduce movement, things change dramatically.
- For example, consider moving the camera:

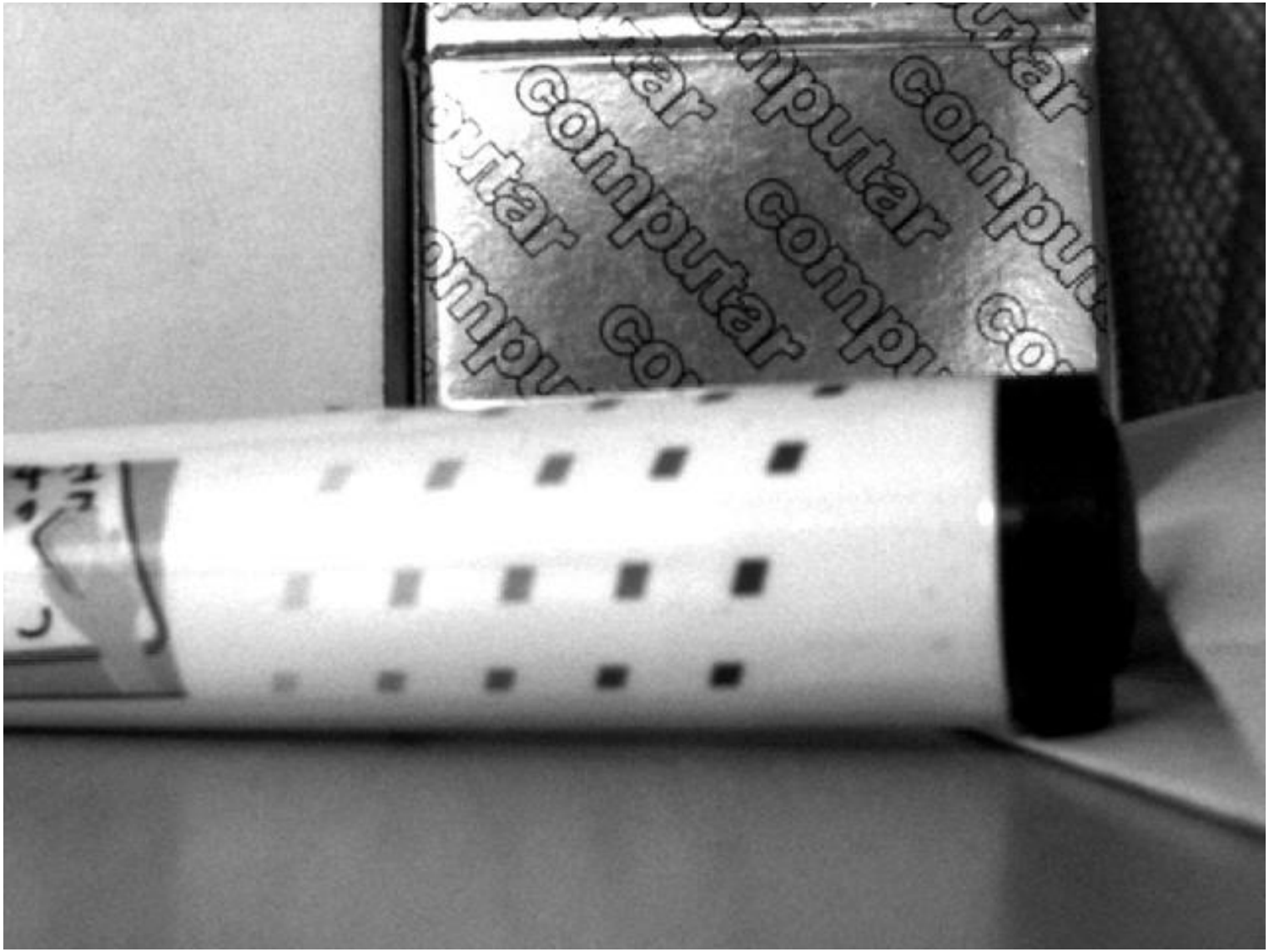




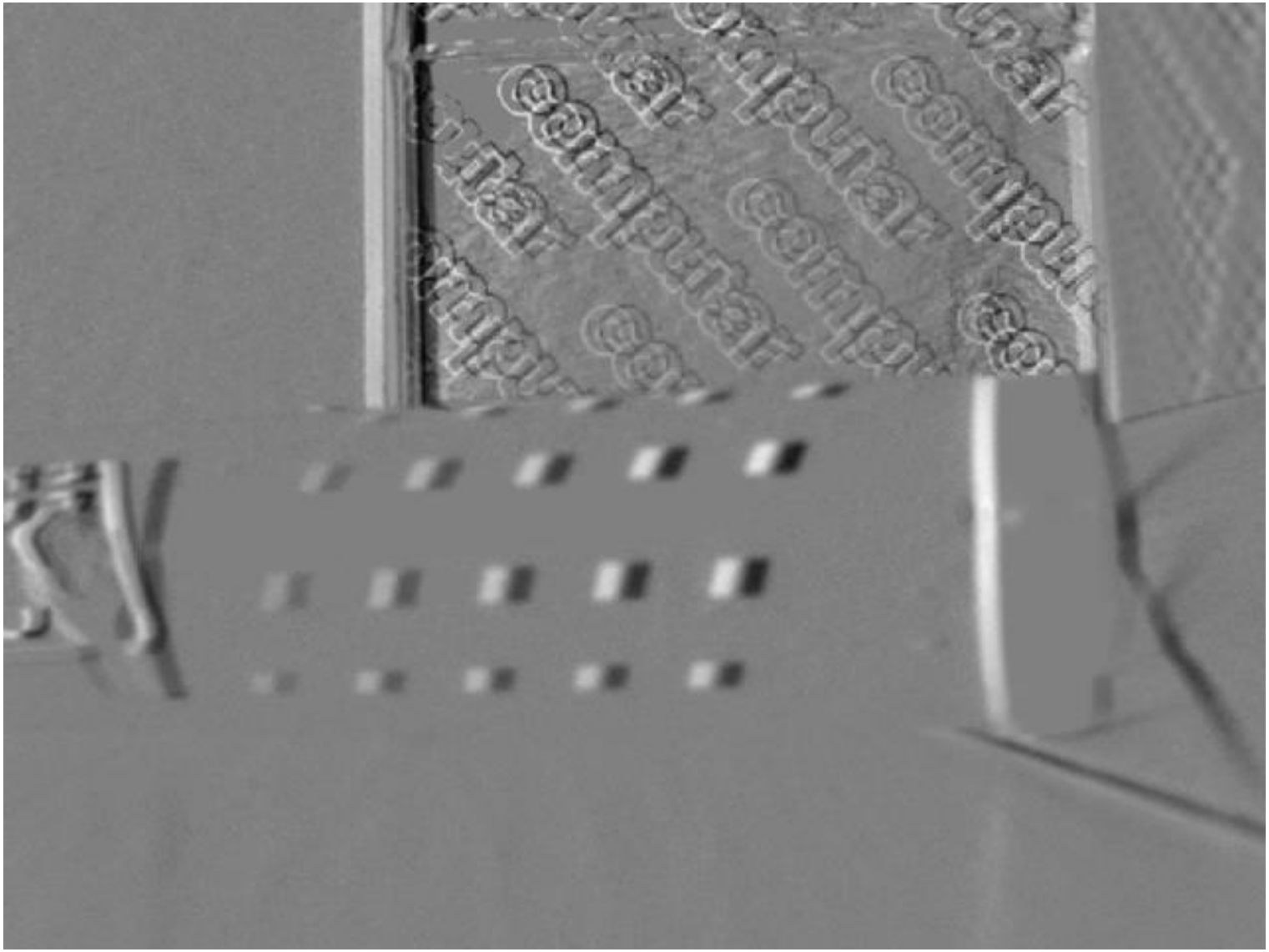
Frame A



Frame B

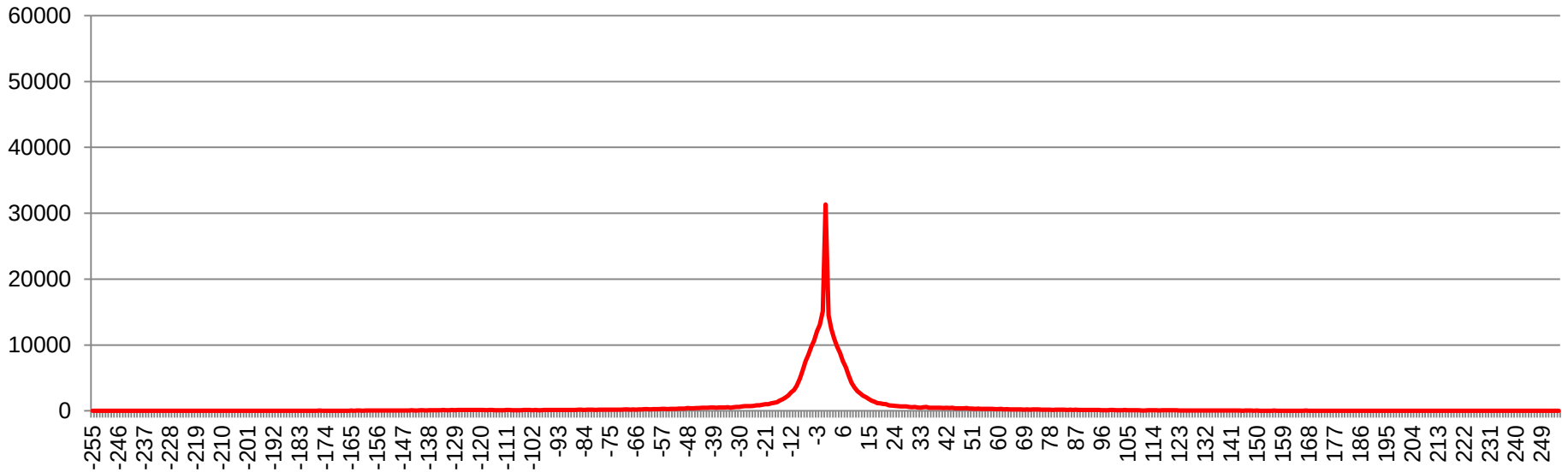


Difference Image



Differences Distribution

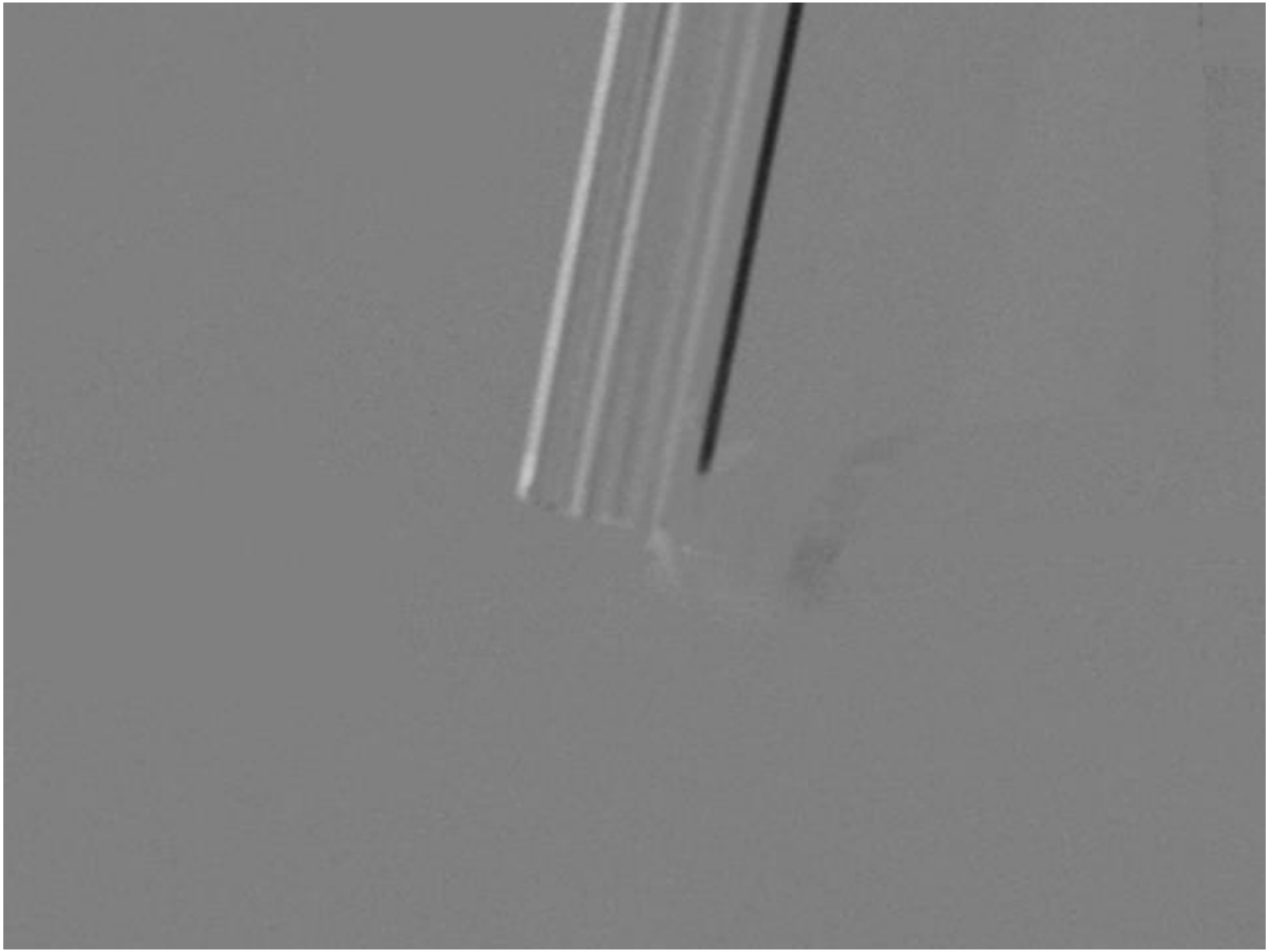
- In this case the situation is very different (H = 6.17).
- The presence of motion introduces a variation that leads the prediction to make a big mistake.
- However, being able to predict the amount of the shift (in vector terms, i.e. direction and modulus) we could take frame A and "move it", to reduce the problem.
- Unfortunately, the same "defect" is found when an object moves in the scene.



Frame A

Frame B

Difference Image



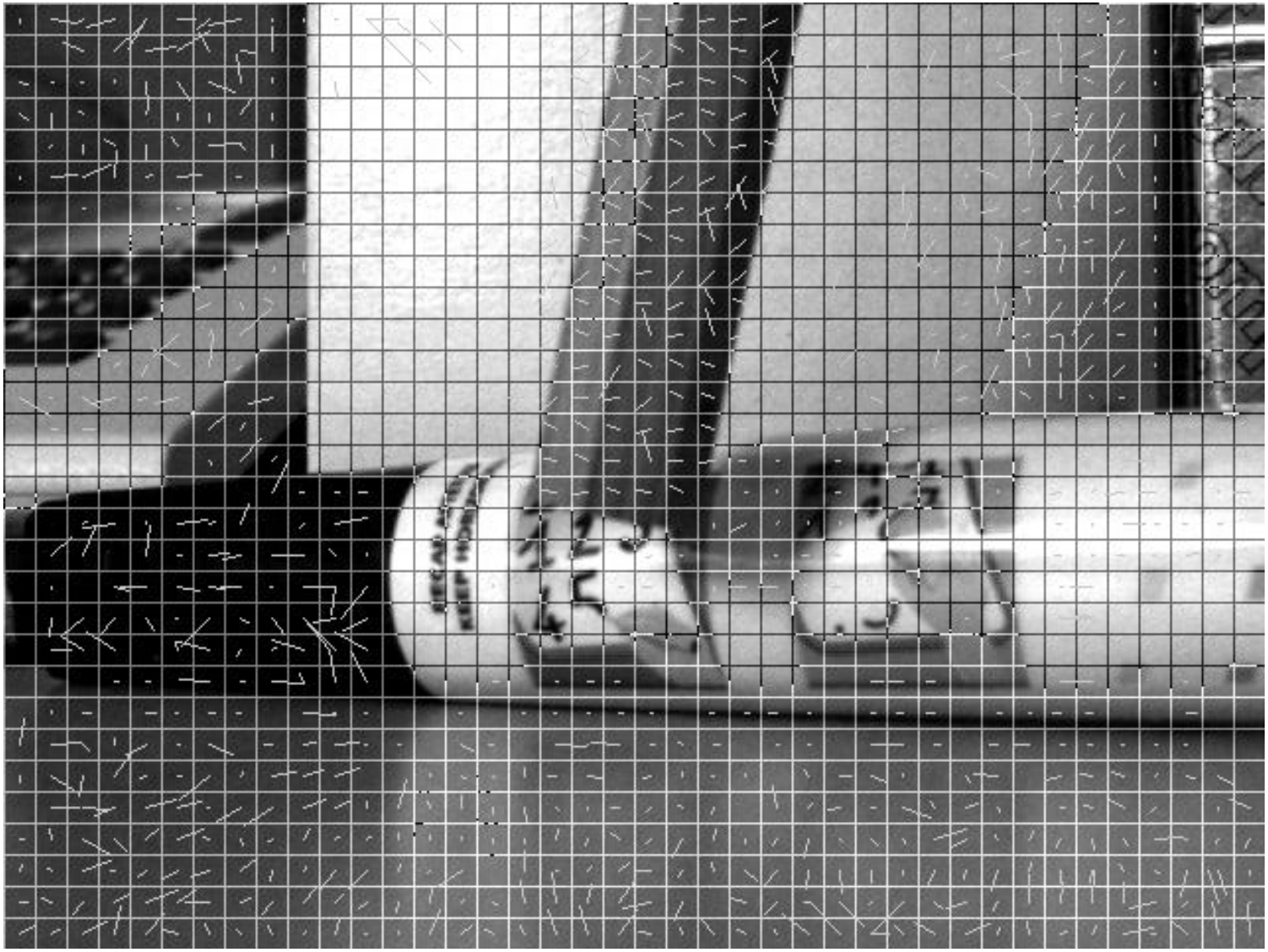
Improve Prediction

- A possible solution to improve the prediction would be to send information on the displacement vector of each pixel, followed by the difference (prediction error).
- However, this is inapplicable, since the displacement vectors would cause an increase in the amount of data, such as to cancel the benefits of the prediction itself.
- For this reason, all video compression standards have used a trade-off strategy: a *motion vector* is sent for groups of pixels.
- 16x16 is the most used block size (H.261, MPEG-1, MPEG-2, MPEG-4). These blocks are called *macroblocks*.
- The choice to use the 16x16 macroblocks is due to the simple correspondence between the color planes. In fact, as seen for JPEG, the color is represented using one luminance component and two chrominance components (YCbCr) and the chrominance components are then sub-sampled with a ½ factor both in height and in width. So every 4 8x8 blocks on Y there are 1 block Cb and 1 block Cr.

Motion Vector

- The Motion Vector is a pair (x, y) that indicates, for the current block, in which position of the previous image the most similar block is located.
- The position, as the name suggests, is indicated as a displacement vector with respect to the current coordinates, therefore x and y can also have negative values.
- Generally, it is not allowed to indicate motion vector that would cause the block of the previous image to be partially outside the image.
- In the context of compression, being "similar" means having small differences. With the aim to minimize the SAD (Sum of Absolute Differences i.e. sum of the absolute values of the differences), a search is carried out.
- There are many techniques for carrying out this search, which produce results more or less close to the optimum, which is obviously easily obtainable by carrying out an exhaustive search (try all possible motion vectors, measure the SAD and keep the one that provides the minimum result) .

Motion Vector of B Compared to A



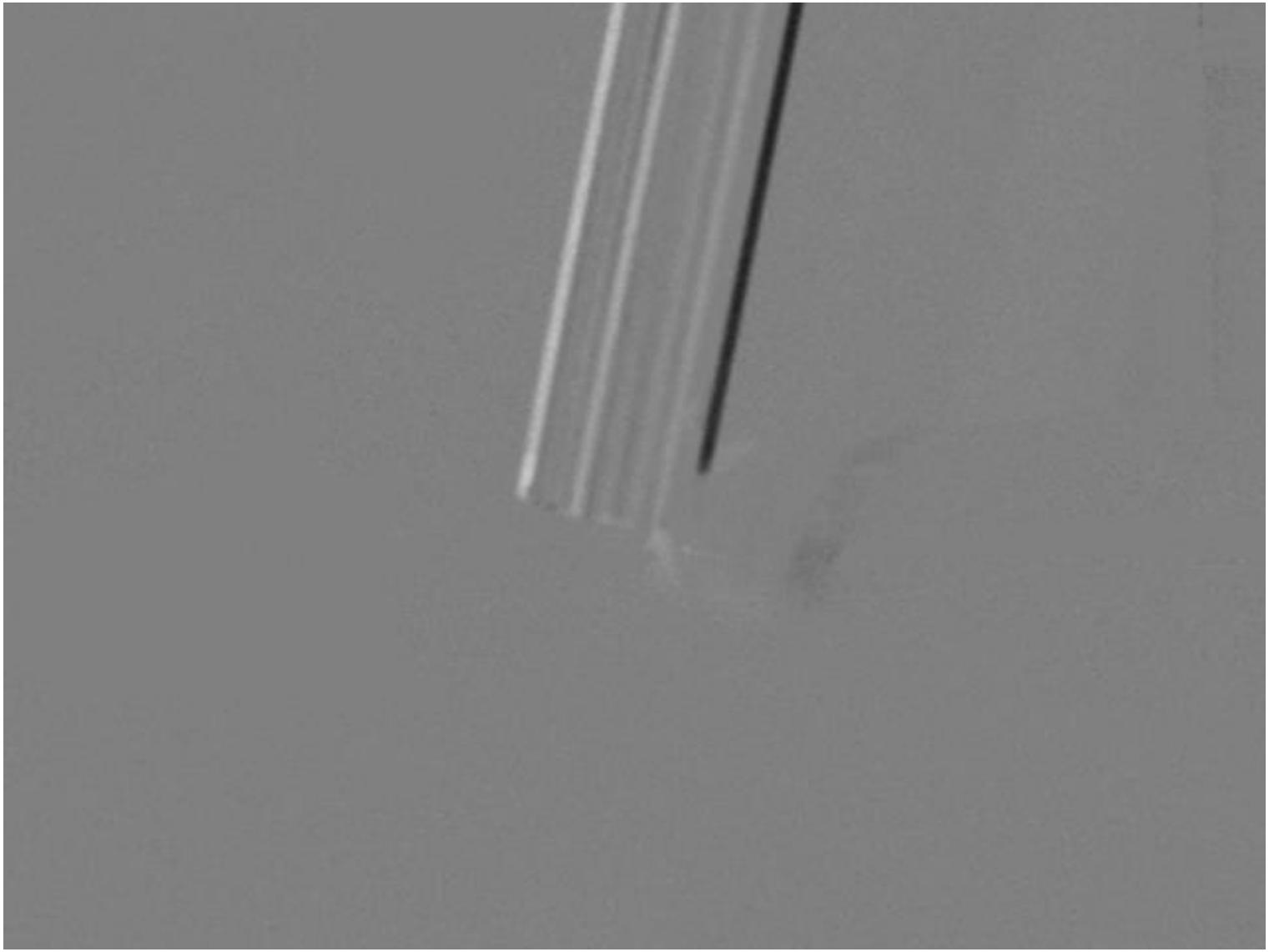
Predicted Image

- By having the previous frame and the motion vector, it is possible to build an image composed of macroblocks of the previous frame, displaced by what is indicated by the motion vector.
- This image (predicted), allows to reconstruct the frame with lower differences and therefore with probable saving of space.
- Unfortunately, the space occupied by motion vectors must also be added. It is therefore necessary to introduce a trade-off between the space saved by reducing the differences and the one "wasted" by the motion vector.
- To do this, for example, it is possible to add a motion vector only if the SAD of the optimal block and that of the corresponding block [MV = (0,0)] have a difference greater than a predetermined threshold.
- It is understood that what is indicated here is only a very trivial example, while the techniques used by real systems can be much more complex.

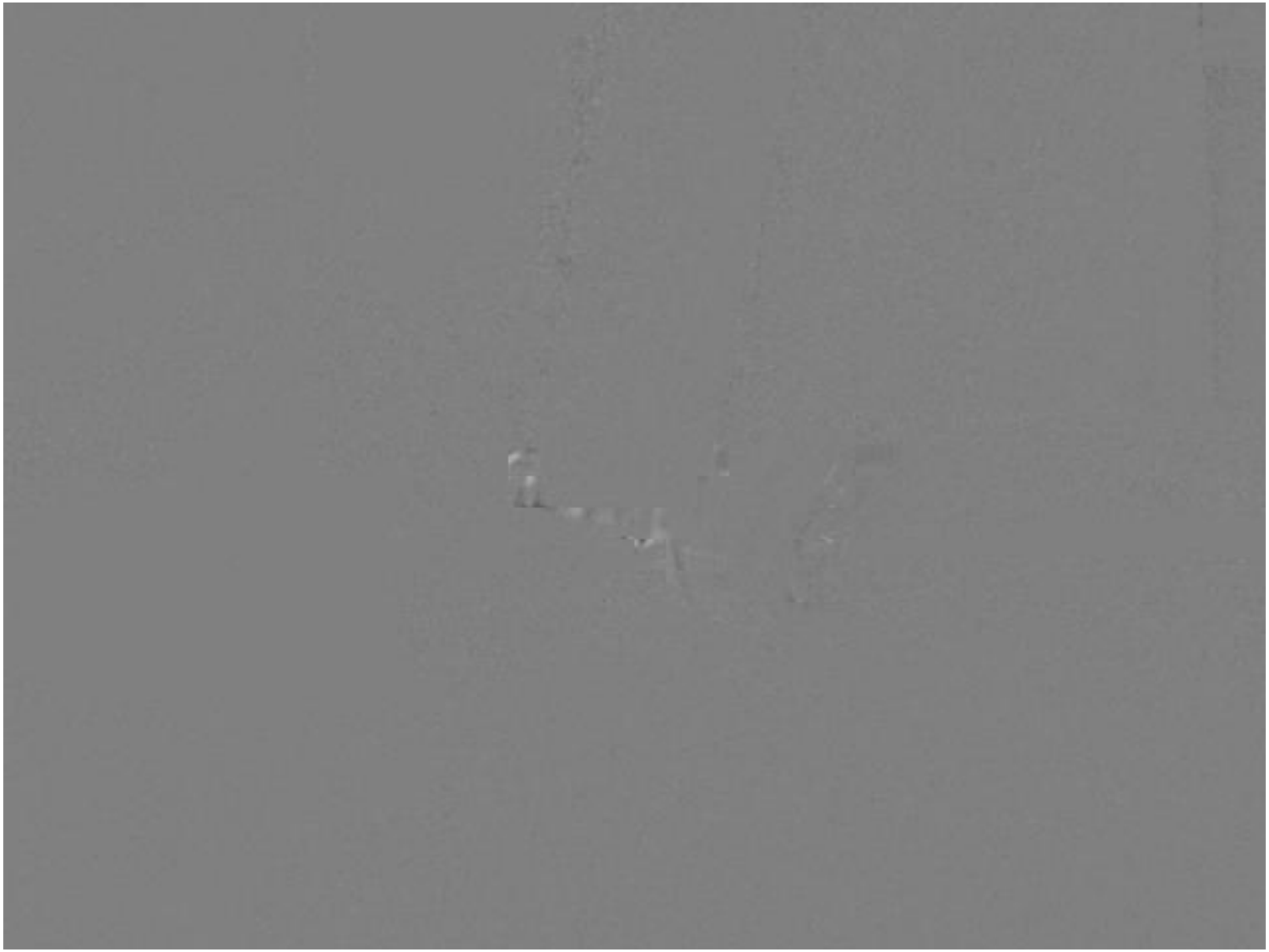
Frame B

Predicted Image (Built With Blocks of A and MV)

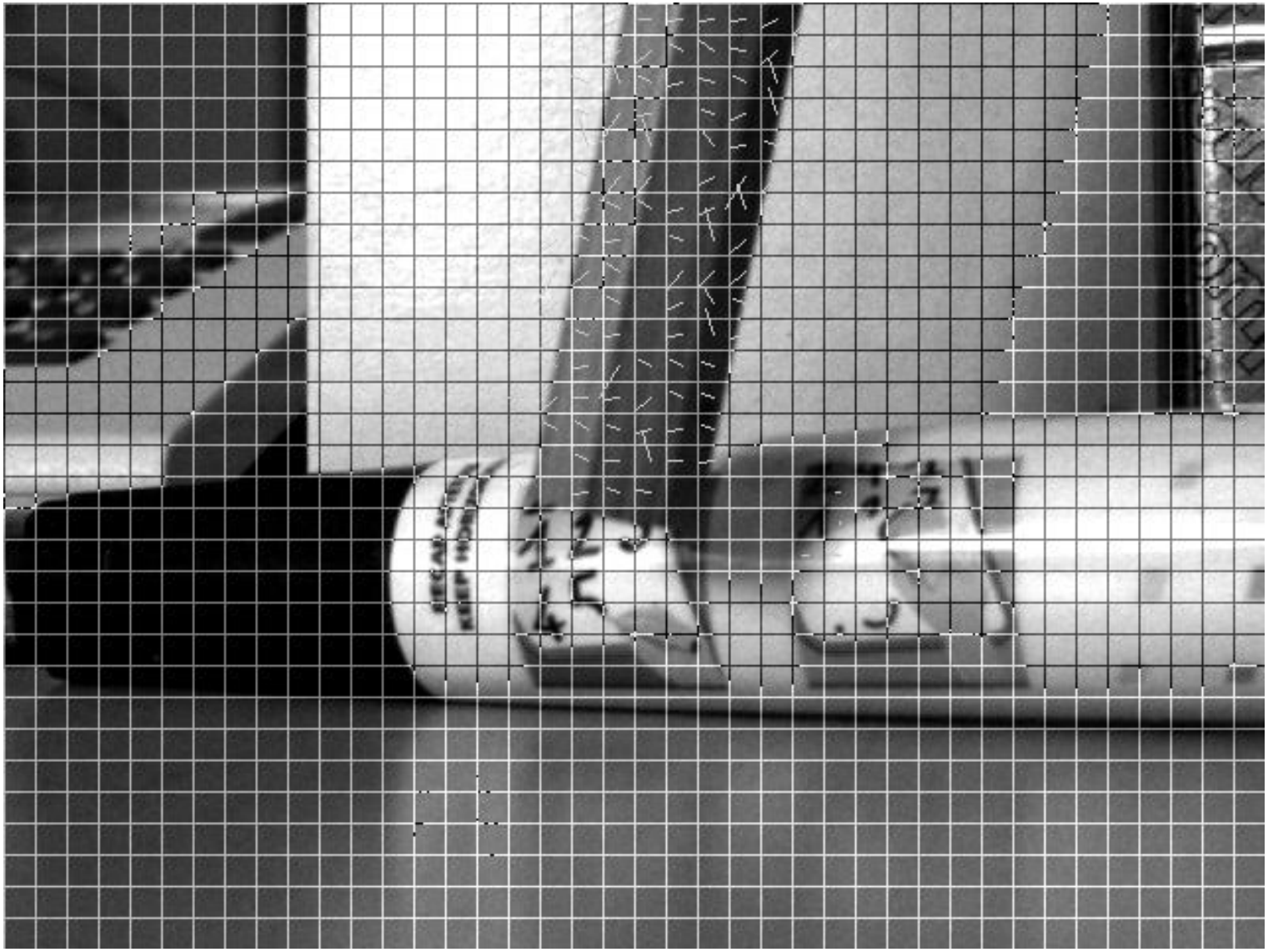
Difference Image (B - A)



Difference Image (B - Prediction of B)



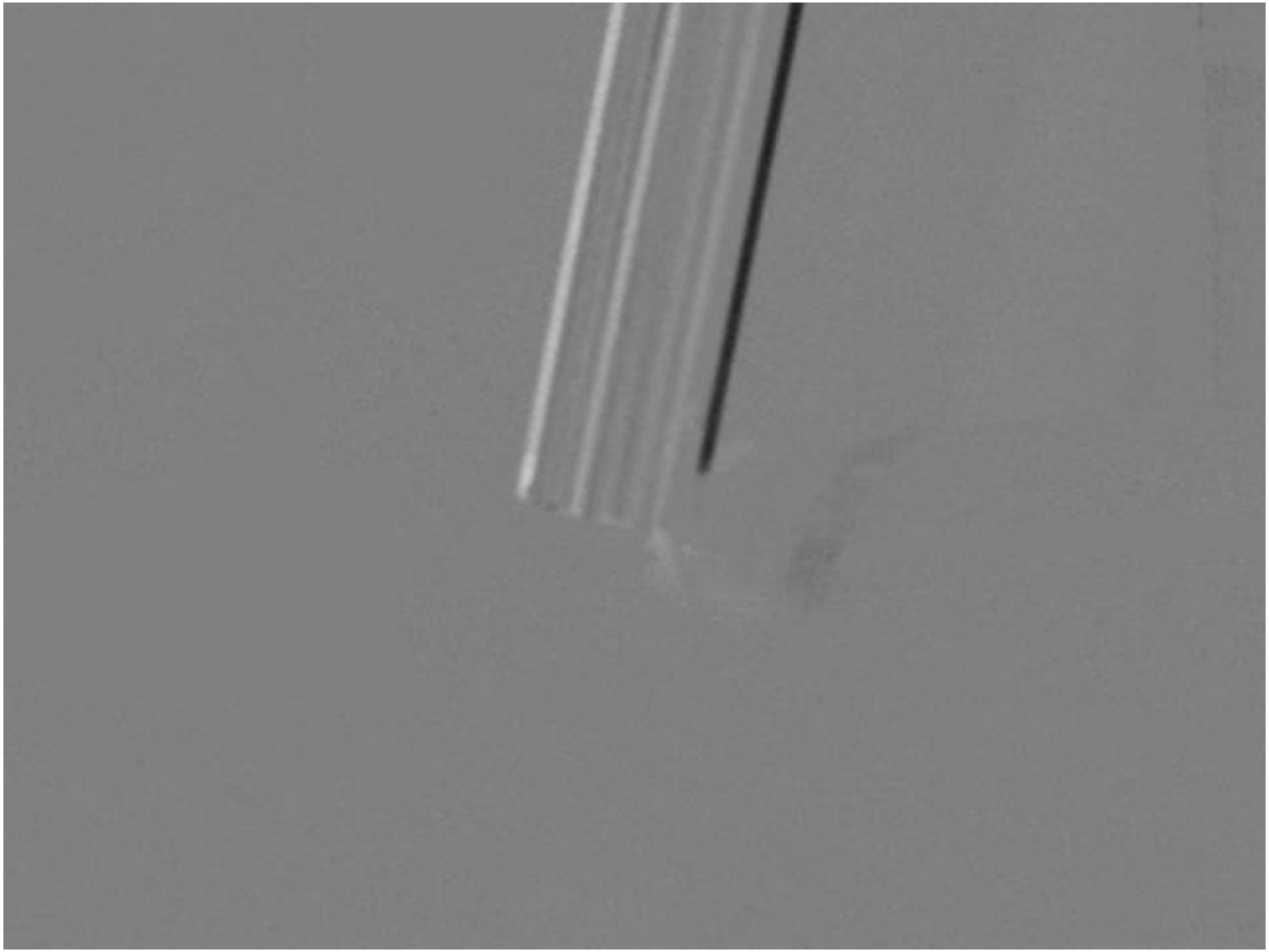
Motion Vector of B compared to A (SAD > 1000)



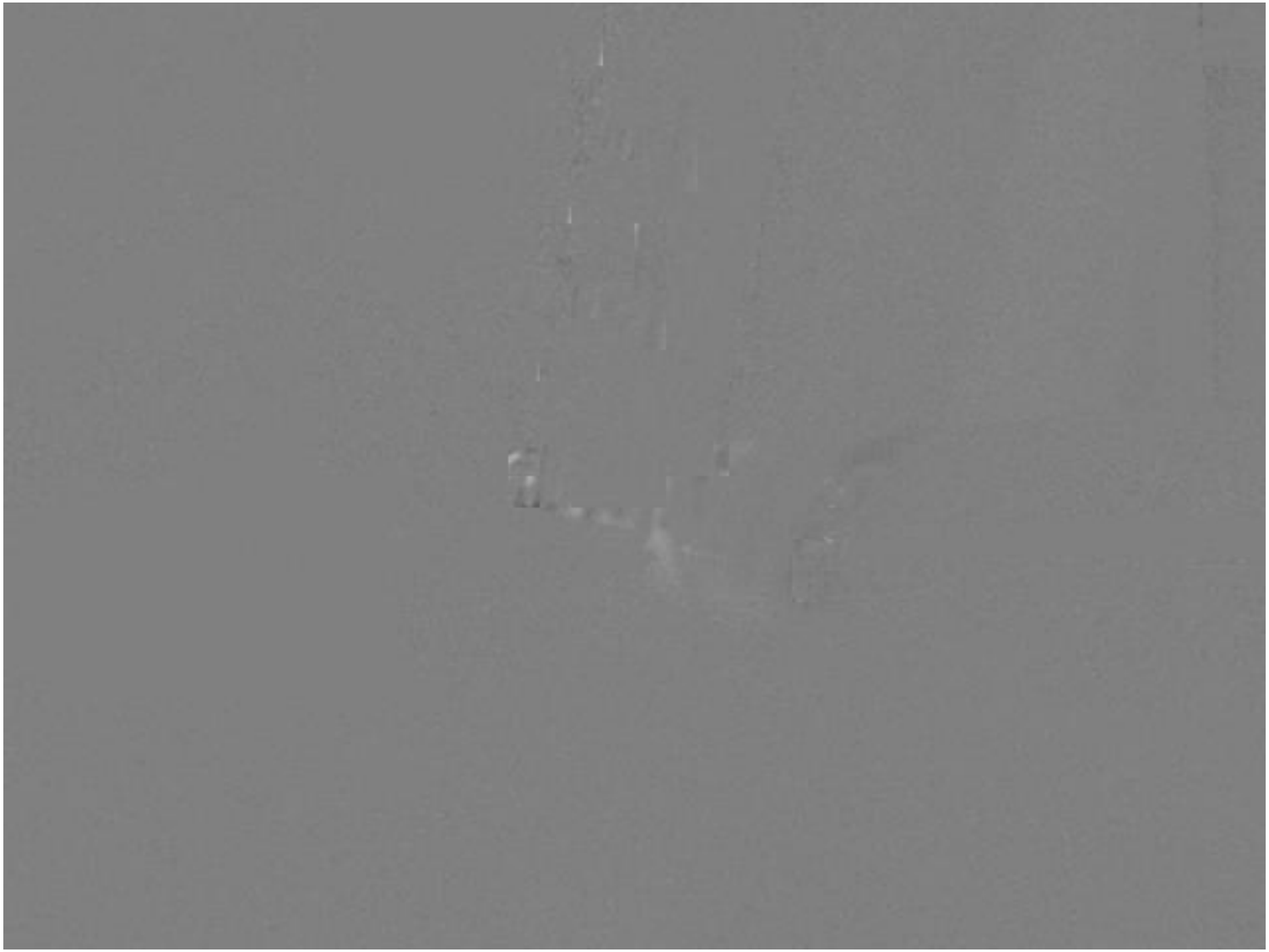
Frame B

Predicted Image (SAD > 1000)

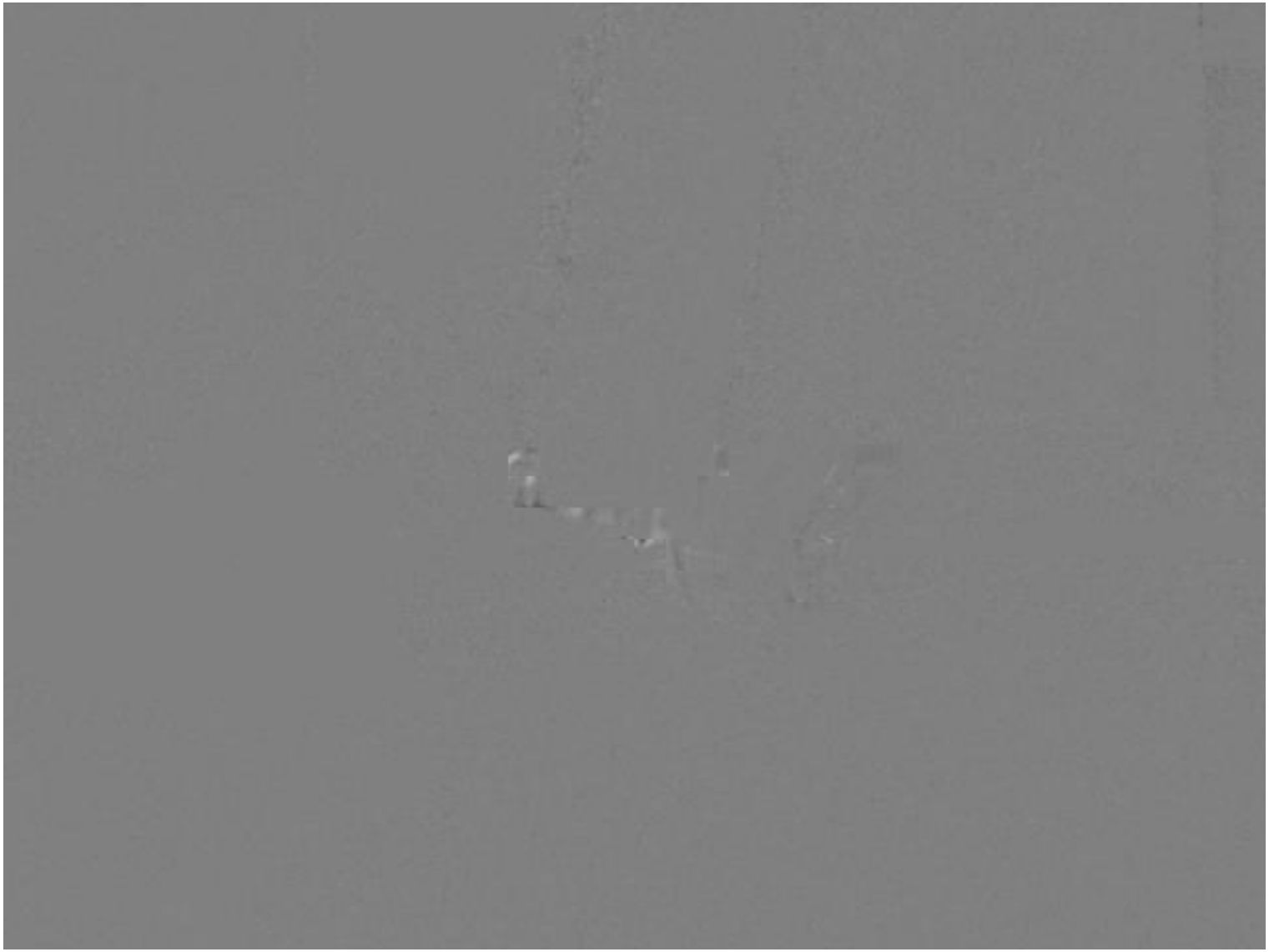
Difference Image (B - A)



Difference Image (B - Pred. B) (SAD > 1000)



Difference Image (B - Pred. B)

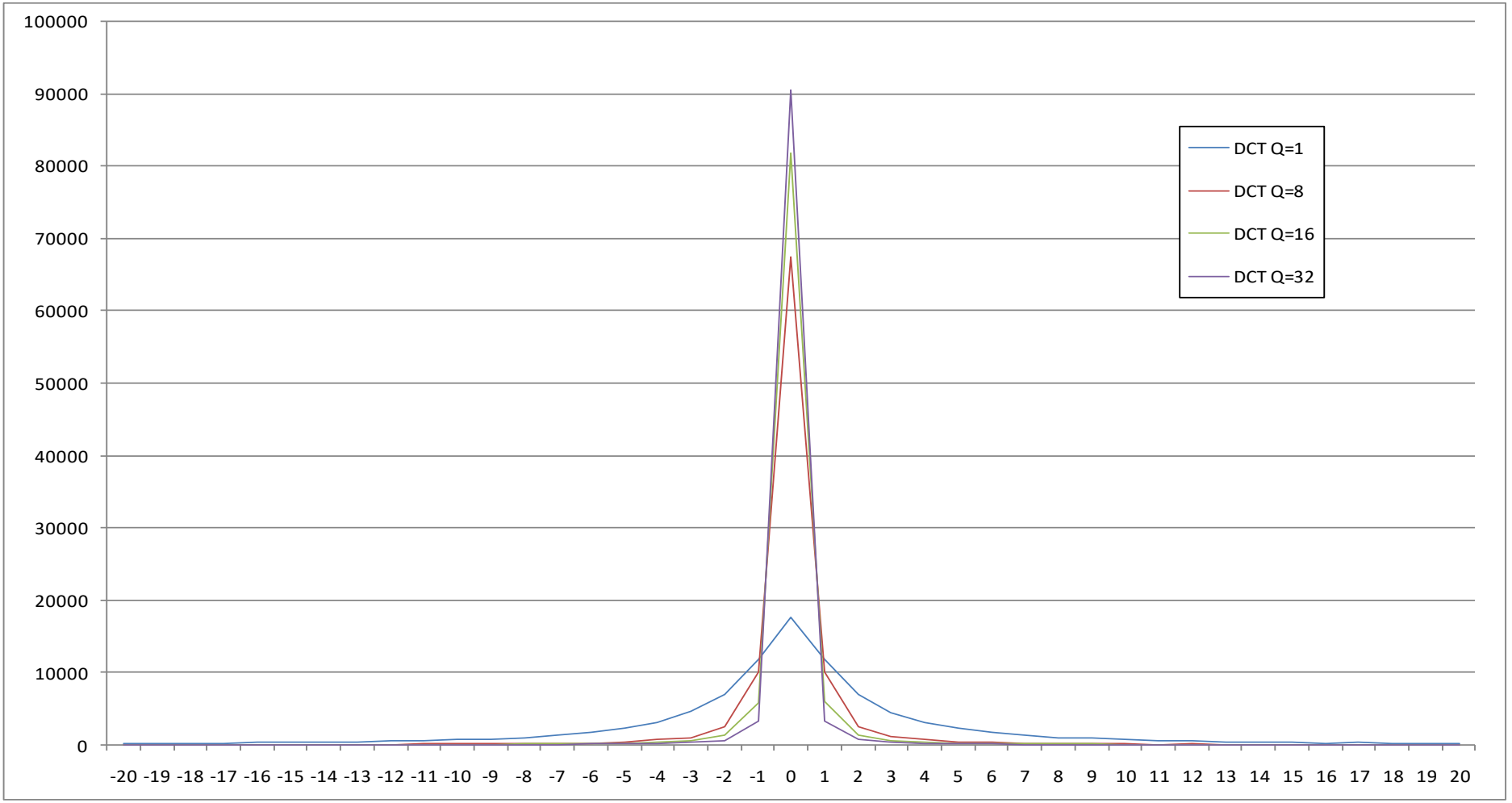


Lossy Compression

- As for image compression (JPEG), not all information must be transmitted unaltered to allow you to appreciate the video content, even with excellent quality.
- It is therefore possible to transmit the pixels encoded as inter frames or the differences encoded as intra frames, accepting a loss in their representation.
- The loss should take place on the less "significant" components of the data. Therefore, we try to represent the data through a transformation that separates the different information contents.
- The main transformation used in video encoding is DCT, followed by a quantization (integer division).

Effect of Quantization on the Coefficients

- Different quantization levels of the DCT coefficients produce drastic reductions in the width (uniformity) of the distribution.



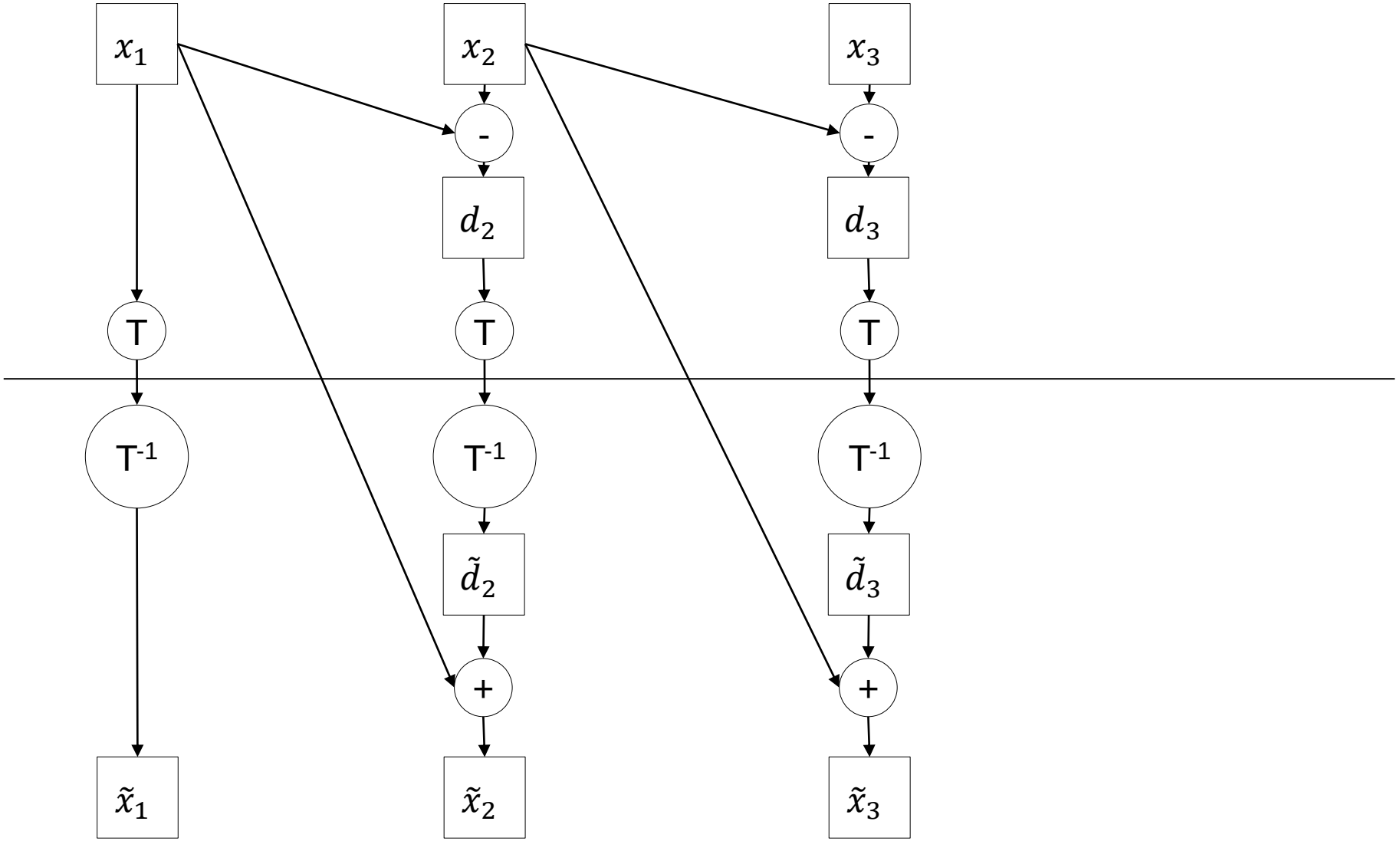
Q = 1 (No Quantization): H=4,91

Q = 8: H=2,10

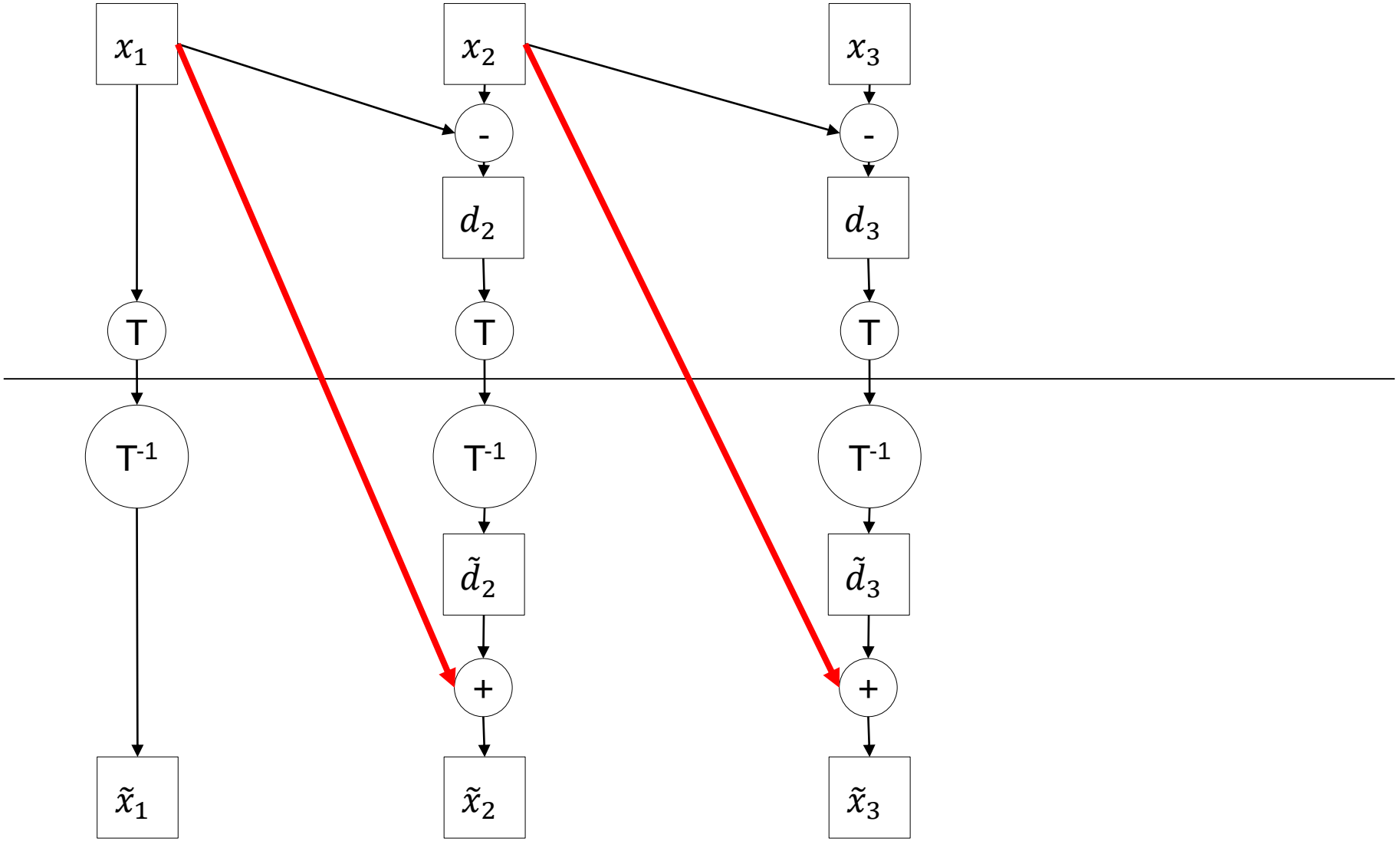
Q = 16: H=1,36

Q = 32: H=0,84

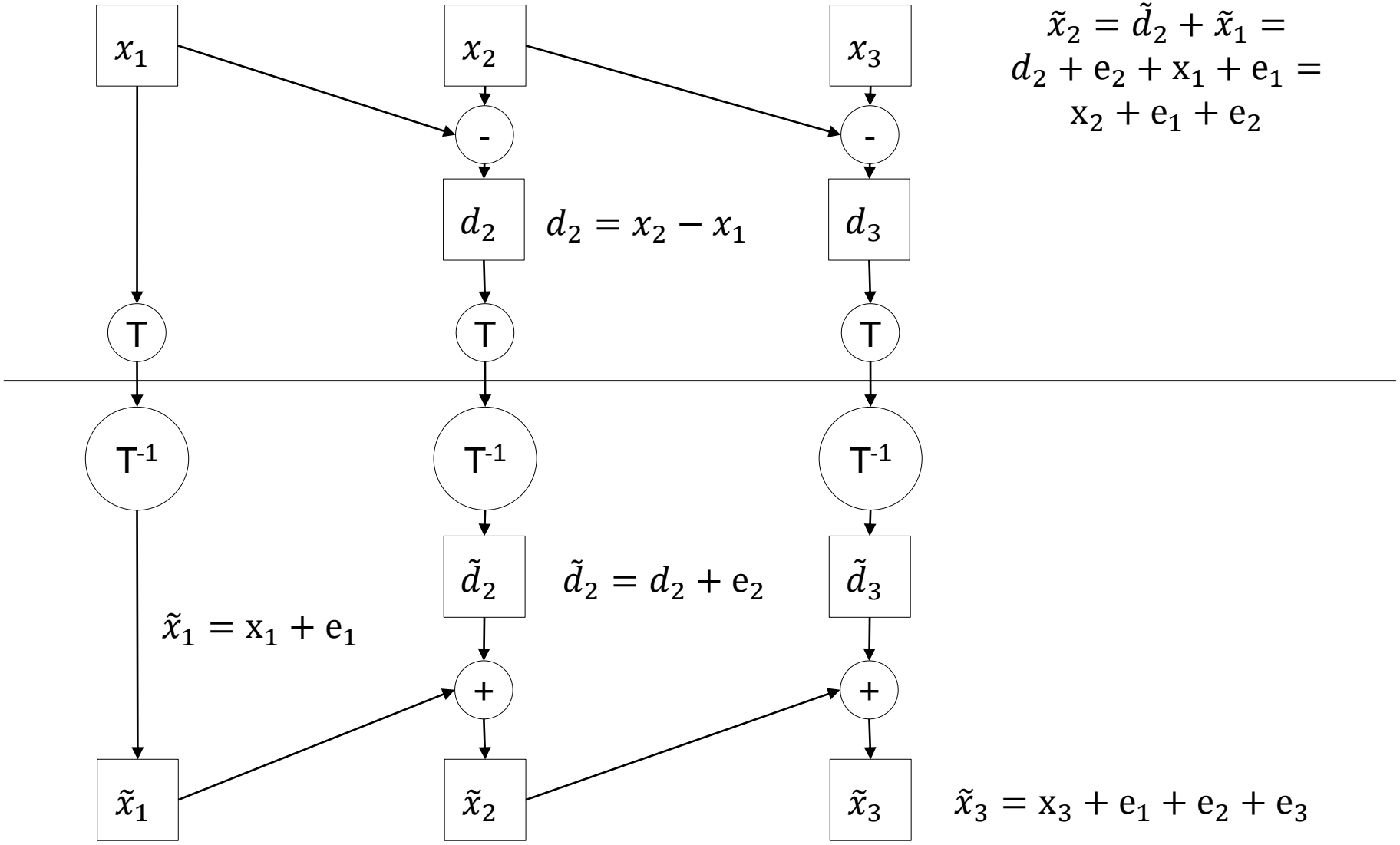
Video compression as I told you until now



Video compression as I told you until now



Video compression as I told you until now

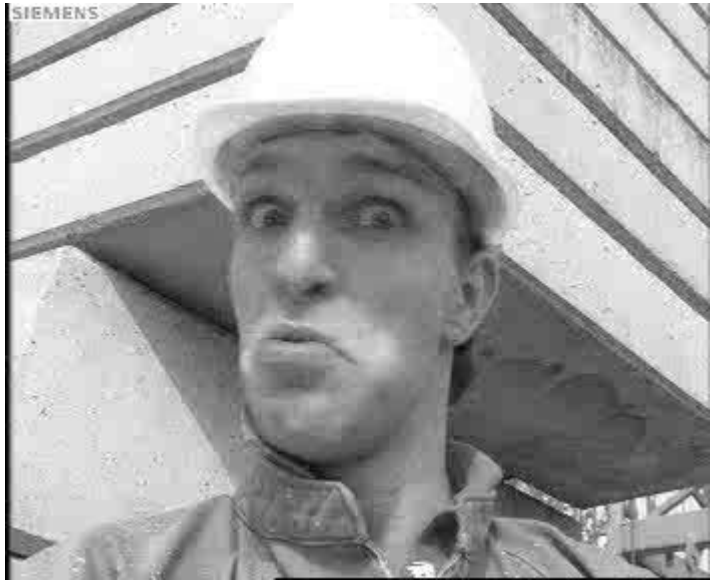


Error Propagation

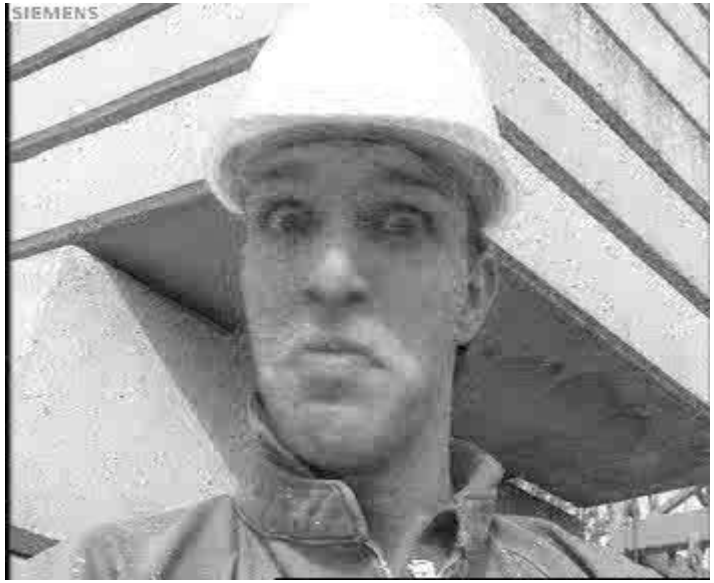
- The approximation of the coefficients during reconstruction introduces a problem for inter frame coding.
- In fact, after the reconstruction, the decoder does not have the original frame, but an approximation of it. For this reason, if the reference is different, the aforementioned image will also be different and therefore the result will be ruined.
- This phenomenon means that after a few frames, the error accumulates, and the reconstruction becomes unacceptable.
- In the following a sequence is shown in which the first frame is compressed inter (DCT 8x8 with uniform quantization at 32), while the others are inter without MV. The differences are always compressed with the indicated parameters.

Example Sequence (Foreman 1/10)

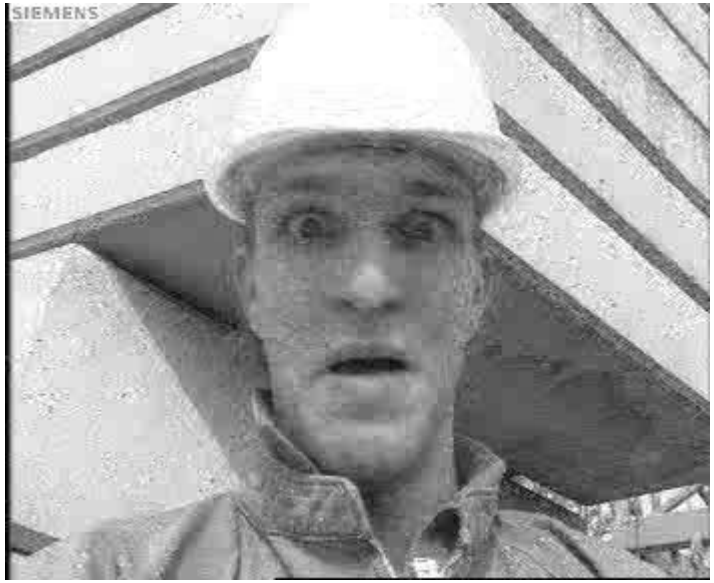
Example Sequence (Foreman 2/10)



Example Sequence (Foreman 3/10)

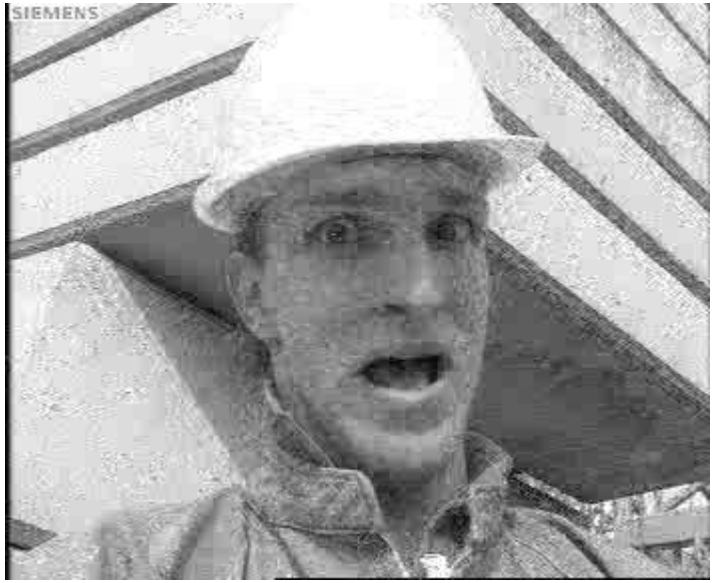


Example Sequence (Foreman 4/10)

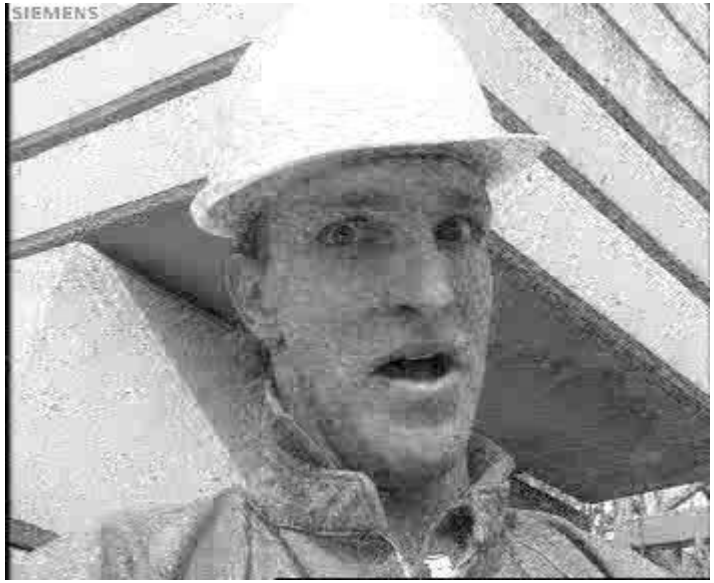


Example Sequence (Foreman 5/10)

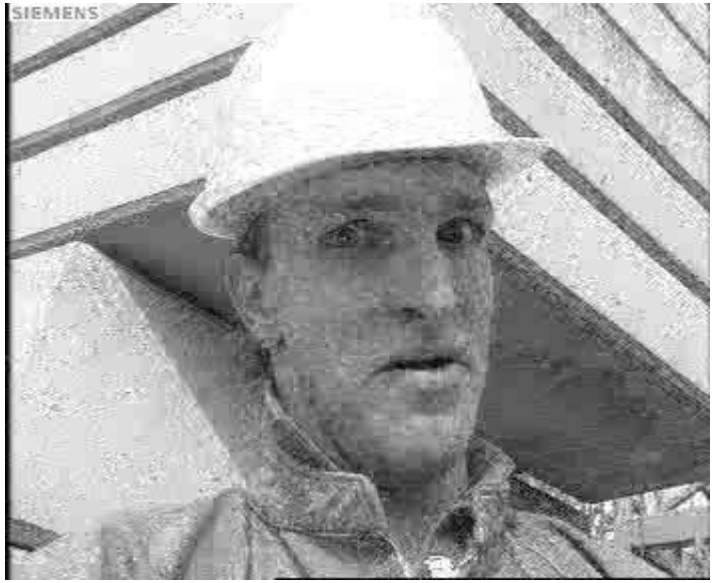
Example Sequence (Foreman 6/10)



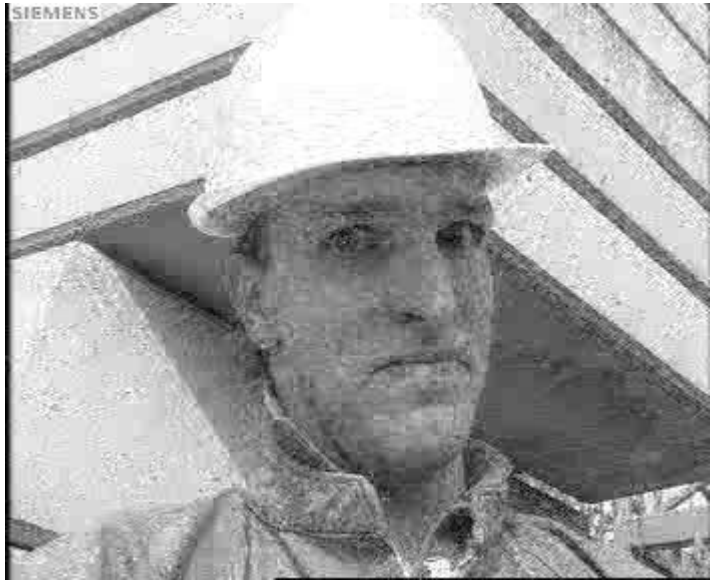
Example Sequence (Foreman 7/10)



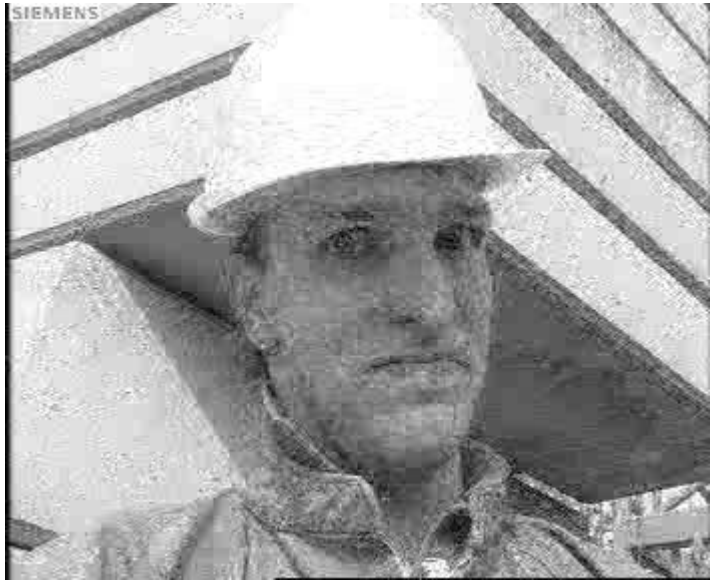
Example Sequence (Foreman 8/10)



Example Sequence (Foreman 9/10)



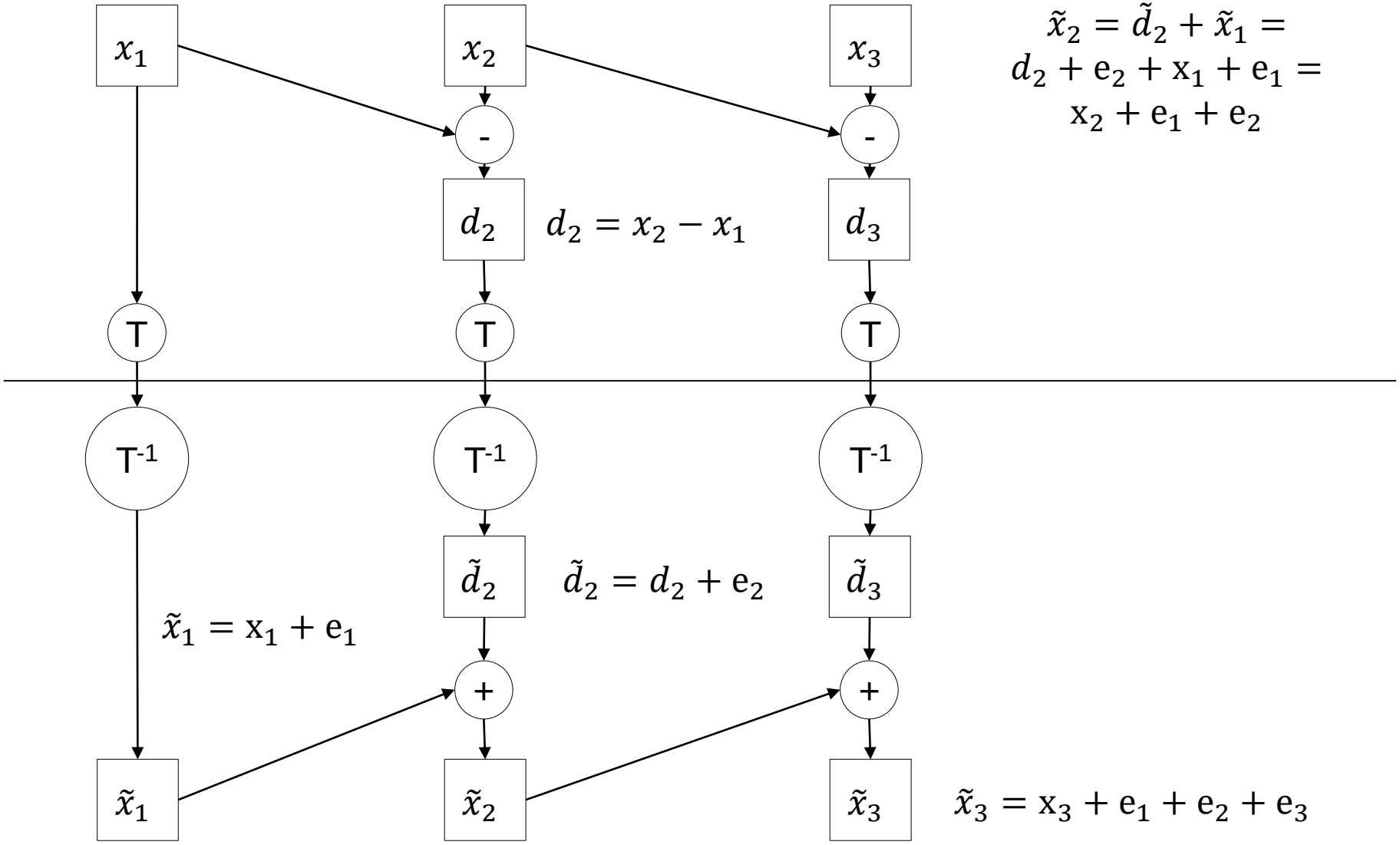
Example Sequence (Foreman 10/10)



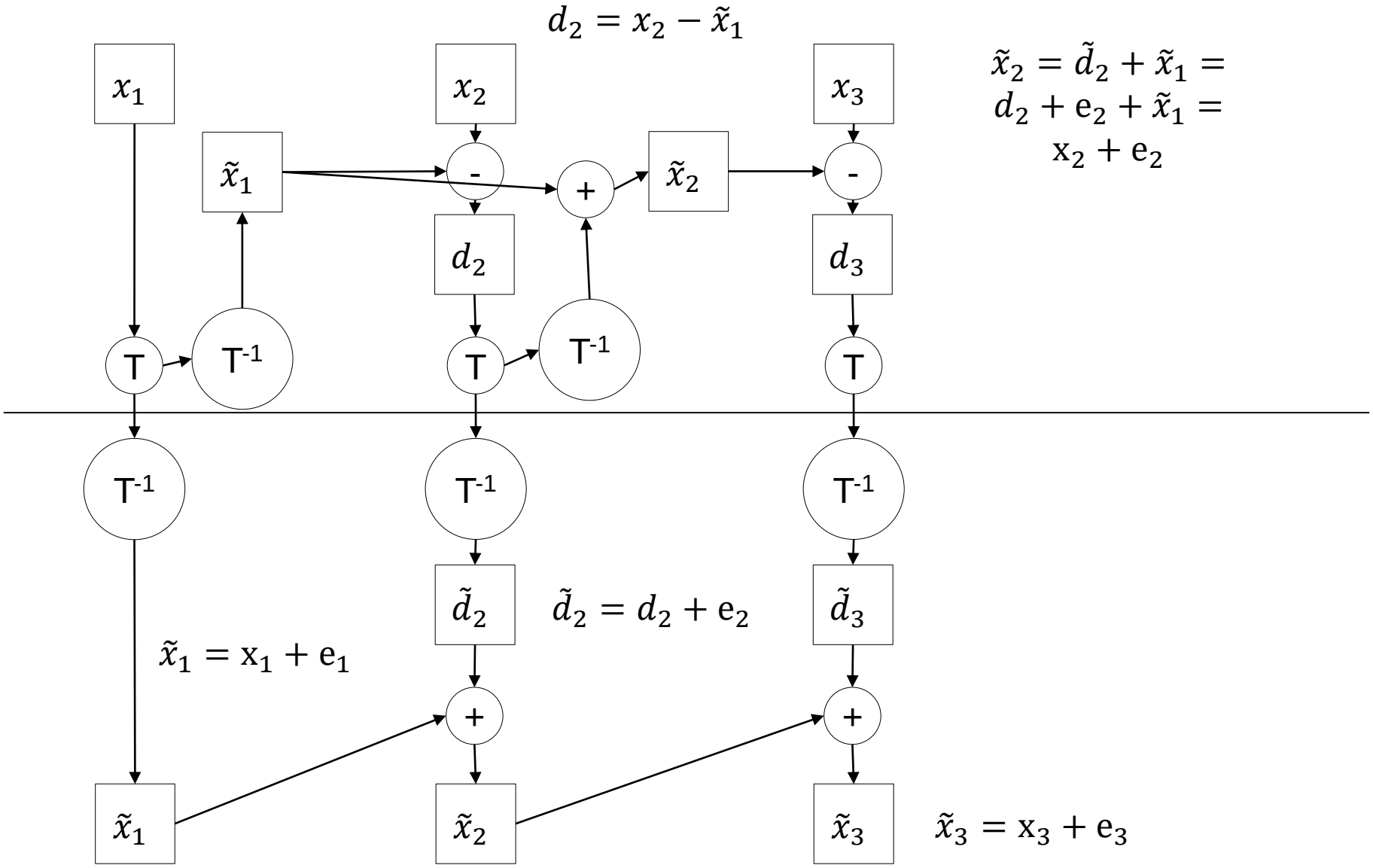
Problem Solution

- The problem is that the decoder does not have the original reference.
- Since it is not possible to send the original reference to the decoder, the only solution is to obtain the frame reconstructed by the decoder also from the encoding side, therefore referring to the same frame.
- This means that after encoding a frame (or the differences), the encoder also performs a decoding, in order to store the uncompressed version of the just compressed data, thus having a reference for the future steps.
- The following sequence is compressed with the same parameters as before, but this time the encoder made the differences with the reconstructed frame.

Video compression as I told you until now

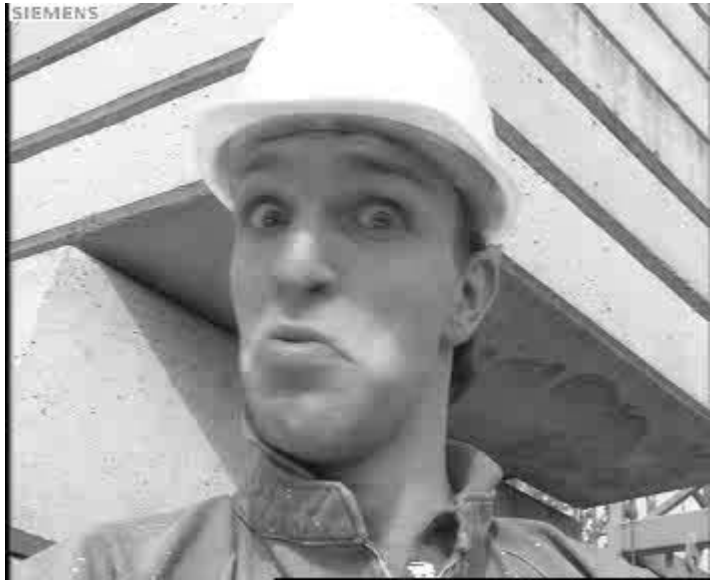


Video compression as it must be done.

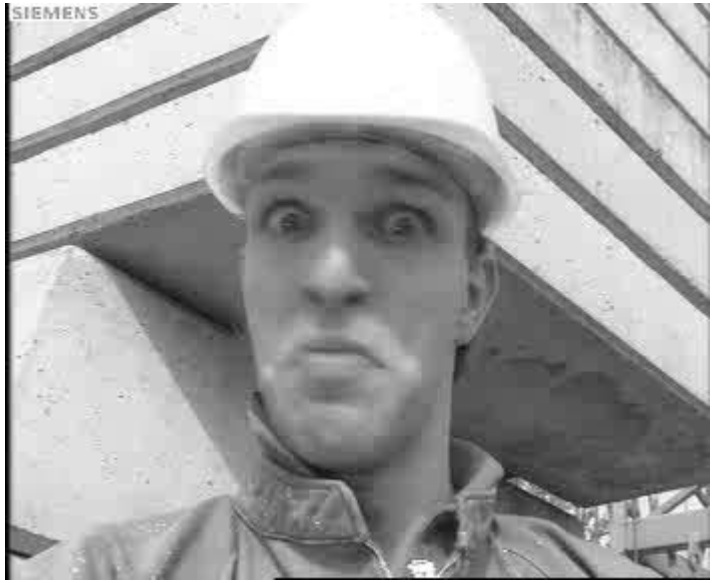


Example Sequence (Foreman 1/10)

Example Sequence (Foreman 2/10)



Example Sequence (Foreman 3/10)



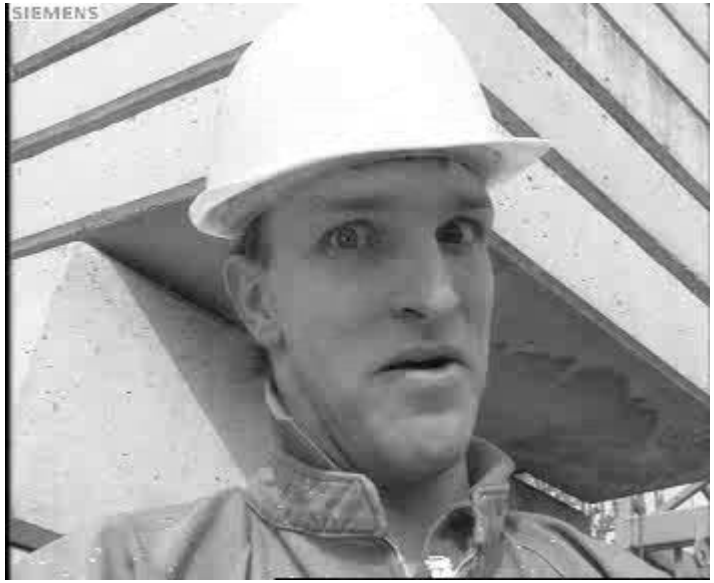
Example Sequence (Foreman 4/10)

Example Sequence (Foreman 5/10)

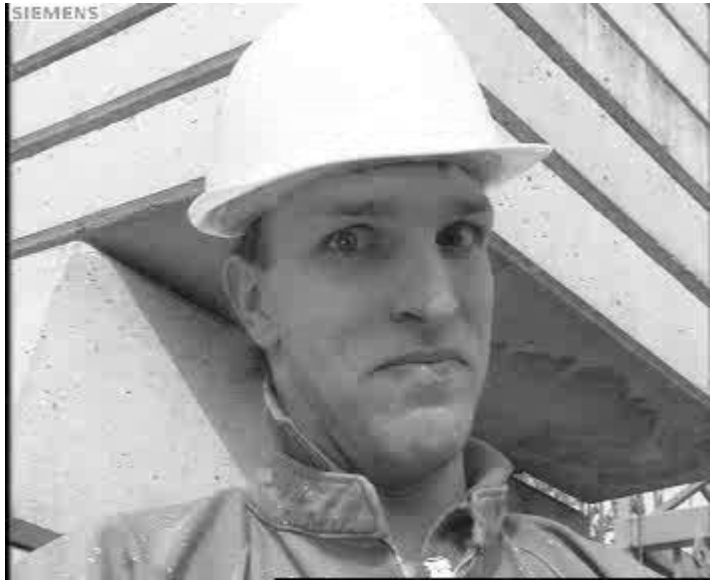
Example Sequence (Foreman 6/10)

Example Sequence (Foreman 7/10)

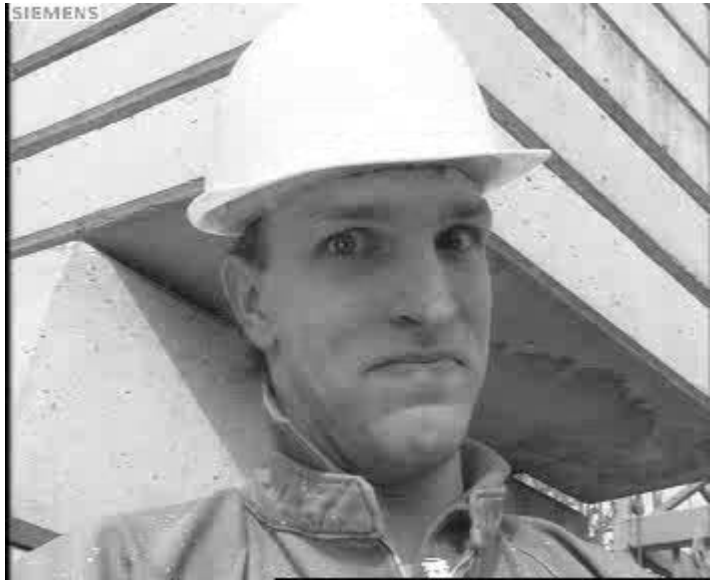
Example Sequence (Foreman 8/10)



Example Sequence (Foreman 9/10)



Example Sequence (Foreman 10/10)



Reference Schema for an Encoder