

Motion Estimation for Video Coding Standards

Prof. Ja-Ling Wu

Department of Computer Science
and Information Engineering
National Taiwan University



1

Motion Estimation for Video Coding Standards

Motion-Compensated estimation is an effective means in reducing the interframe correlation for image sequence coding.

H.261/263 , ISO MPEG-1/2/4, H.264/AVC
motion estimation techniques are also used in many
other applications :


Computer Vision , target tracking , industrial monitoring

Remark :

「 The goal of image compression is to reduce the total transmission bit rate for reconstructing images at the receiver. Hence, the motion information should occupy only a small amount of the transmission bandwidth additional to the picture contents information. →



2




As long as the motion parameters we obtain can effectively reduce the total bit rate, these parameters need not be the true motion parameters. →

Drift Problem


If the reconstructed images are used for estimating motion information, a rather strong noise component can not be neglected. 」

Notice that the current video standards specify only the decoder, i.e. the encoder which performs the motion estimation operation is not explicitly specified in the standards →

a great amount of flexibility exists in choosing and designing a motion estimation scheme for a standard coder.



3




Motion estimation techniques can be classified, roughly, into three groups :

- (i) Block matching methods
- (ii) Differential (Gradient) methods
- (iii) Fourier Methods


Definitions :

Motion estimation is the operation that estimates the motion parameters of moving objects in the image sequence.

Motion compensation is the operation of predicting a picture, or portion thereof, based on displaced pels of a previously transmitted frame in an image sequence.



4

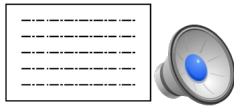


In the physical world, objects are moving in the 4-D spatio-temporal domain – three spatial coordinates and one temporal coordinate. However, images taken by a single camera are the projections of these 3-D spatial objects onto the 2-D image plane.→


If pels on the image plane (the output from an ordinary camera) are the only data we can collect, the information we have is a 3-D cube – two spatial coordinates and one temporal coordinate.

In reality, video taken by an ordinary camera is first sampled along the temporal axis. Typically, the temporal sampling rate (frame rate) is 24.25 or 29.97 frames/sec. Spatially, every frame is sampled vertically into a number of horizontal lines, a line is sampled into a number of pels.

(240×352)




5




- To reduce computation and storage complexity, motion parameters of objects in a picture are estimated based on two or three nearby frames.
- general assumptions:
 - (i) Objects are rigid bodies; hence, object deformation can be neglected for at least a few nearby frames.
 - (ii) Objects move only in translational movement for, at least, a few frames.
 - (iii) Illumination are unchanged under movement.
 - (iv) Occlusion of one object by another and uncovered background are neglected.

motion estimation →

- motion segmentation (identify the moving object boundaries)
- motion parameter estimation




6




- Moving object:
A group of contiguous pels that share the same set of motion parameters, it does not necessarily match the ordinary meaning of object.
still background can be considered as a single object
- The effect of object size:
small objects(or evaluation windows)

Ambiguity problem:
Similar objects (image patterns) may appear at multiple locations inside a picture and may lead to incorrect displacement vectors.

Noise sensitivity problem:
Statistically, estimates based on a small set of data are more vulnerable to random noise than those based on a large set of data.




7



large objects(or evaluation windows)

Accuracy problem:
pels inside an object or evaluation window do not share the same motion parameters and, therefore, the estimated motion parameters are not accurate for some or all pels in it.
one must first know precisely the moving object boundaries. → Object Segmentation



8

■ Practical solution:

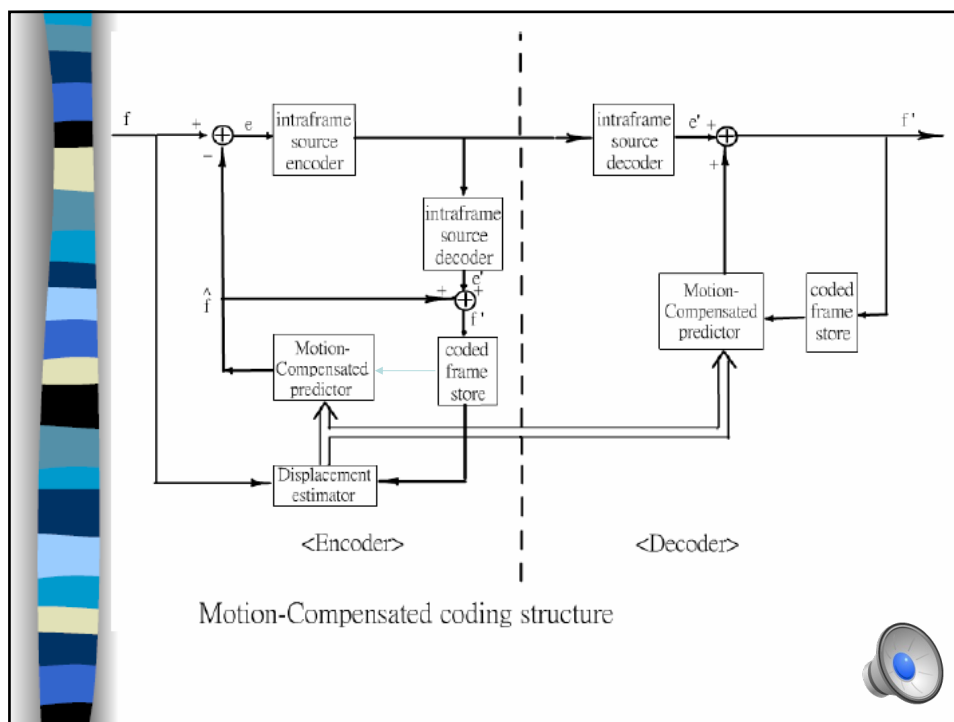
"Partition images into regular, non-overlapped blocks; assuming that moving objects can be approximated reasonably well by regular shaped blocks. Then, a single displacement vector is estimated for the entire image block under the assumption that all the pels in the block share the same displacement vector.

The above assumption may not always be true because an image block may contain more than one moving object. In image sequence coding, however, prediction errors due to imperfect motion compensation are coded and transmitted"

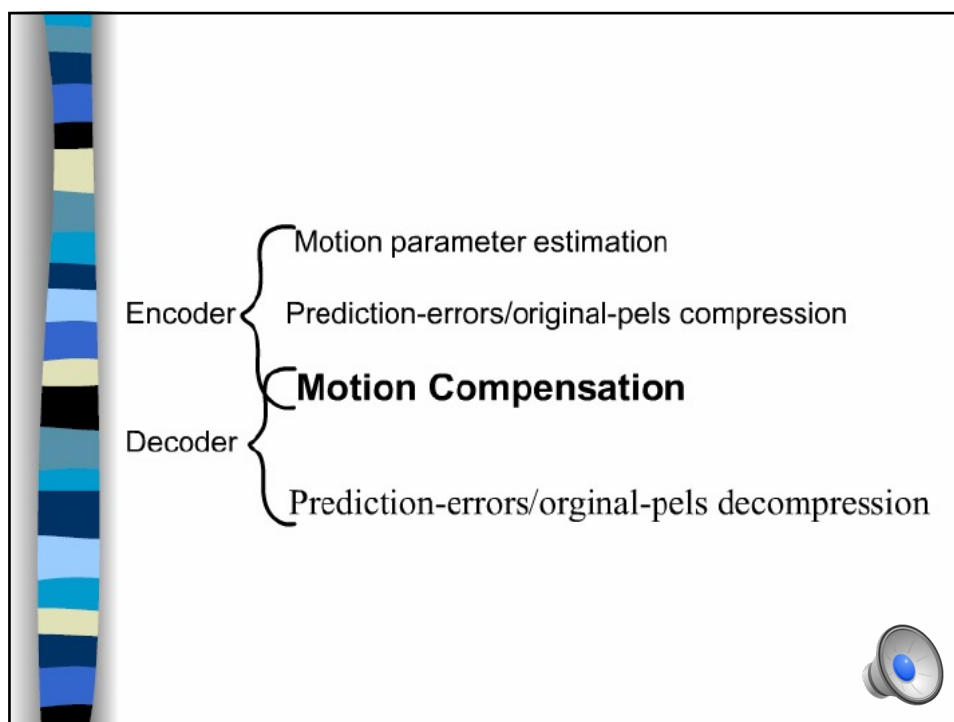
The block-based motion estimation approach is adopted by the video coding standards for partially, at least, its robustness as compared to the pel-based approach.



9



10



11

■ **Motion Compensation:**

- (1) Access the previous-frame image data according to the estimated displacement vectors.
- (2) Construct the predicted pels by passing the previous-frame image through the prediction filter.

H.261/263 : I,P

MPEG-1/-2 : I,P,B \Rightarrow encoding/transmission order

*A picture frame in MPEG-2 may contain two interleaved fields

\Rightarrow frame-based Motion compensation

field-based

A speaker icon is located in the bottom right corner.

12

■ Block Matching Method

Jain & Jain, IEEE Trans. Communications, vol.COM-29, pp.1799-1809, Dec. 1981

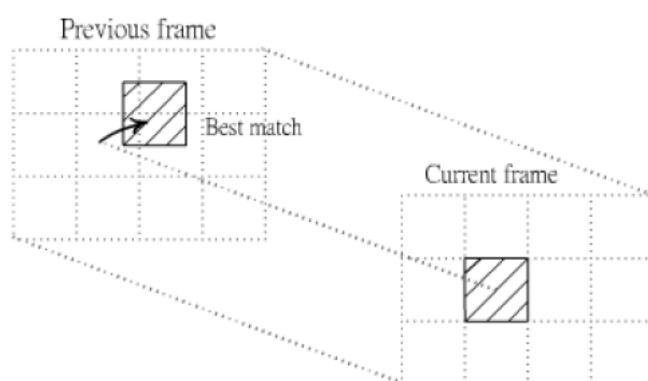
“Displacement Measurement and its application in interframe image coding, ” → Block matching

- reduce the computational load in calculating the motion vector.
- Increase the motion vector accuracy.

Block matching is a correlation technique that searches for the best match between the current image block and candidates in a confined area of the previous frame.




13



Images are partitioned into non-overlapped rectangular blocks. Each block is viewed as an independent object and it is assumed that the motion of pels within the same block is uniform.



14



■ Remarks:


The size of the block affects the performance of motion estimation.

Small block sizes afford good approximation to the natural object boundaries; they also provide good approximation to real motion, which is now approximated by piecewise translation movement. However, small block sizes produce a large amount of raw motion information, which increases the number of transmission bits or the required data compression complexity to condense this motion information.


From performance point of view, small blocks also suffer from object (block) ambiguity problem and the random noise problem.

Large block may produce less accurate motion vectors, since a large block may likely contain pels moving at different speeds and directions.

8×8,16×16 : H.261, MPEG-1,MPEG-2




15



- The basic operation of block matching is picking up a candidate block and calculating the matching function (usually a nonnegative function of the intensity differences) between the candidate and the current block.
- This operation is repeated until all the candidates have gone through and then the best matched candidate is identified. The location of the best matched candidate becomes the estimated displacement vector.

■ Important parameters:

- (i) the number of candidate blocks, search points.
- (ii) the matching function.
- (iii) the search order of candidates.



16

■ Search range:

- the maximum range of motion vectors.
- Decided by experiment or due to hardware constraints.

Assume that the size of the image block is $N_1 \times N_2$ and the maximum horizontal and vertical displacements are less than $d_{\max-x}$ and $d_{\max-y}$, respectively.

Assume only integer-value motion vectors are considered:

The size of the search range = $(N_1 + 2d_{\max-x})(N_2 + 2d_{\max-y})$

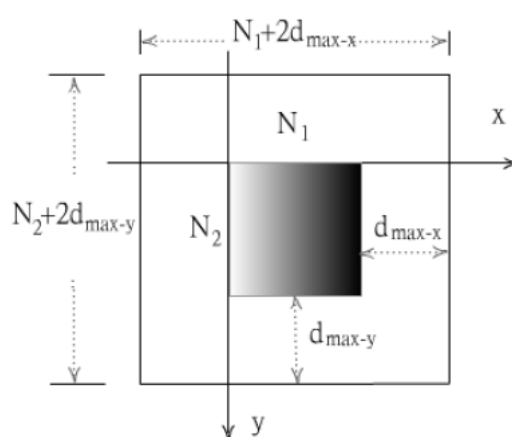
The number of candidate blocks = $(2d_{\max-x} + 1)(2d_{\max-y} + 1)$

exhaustive search :

computational load $\propto (d_{\max-x} \cdot d_{\max-y})$



17



Search range in block matching



18

■ Matching Function

The selection of the matching function has a direct impact on the computational complexity and the displacement vector accuracy.

Let (d_1, d_2) represent a motion vector candidate inside the search region and $f(n_1, n_2, t)$ be the digitized image intensity at the integer-valued 2-D image coordinate (n_1, n_2) of the t -th frame.

1. Normalized cross-correlation function (NCF):

$$NCF(d_1, d_2) = \frac{\sum_{n_1} \sum_{n_2} f(n_1, n_2, t) f(n_1 - d_1, n_2 - d_2, t-1)}{[\sum_{n_1} \sum_{n_2} f^2(n_1, n_2, t)]^{\frac{1}{2}} [\sum_{n_1} \sum_{n_2} f^2(n_1 - d_1, n_2 - d_2, t-1)]^{\frac{1}{2}}}$$



19

2. Mean square error (MSE):

$$MSE(d_1, d_2) = \frac{1}{N_1 N_2} \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} [f(n_1, n_2, t) - f(n_1 - d_1, n_2 - d_2, t-1)]^2$$

3. Mean absolute difference (MAD):

$$MAD(d_1, d_2) = \frac{1}{N_1 N_2} \sum_{n_1} \sum_{n_2} |f(n_1, n_2, t) - f(n_1 - d_1, n_2 - d_2, t-1)|$$

4. Number of threshold differences (NTD):


$$NTD(d_1, d_2) = \sum_{n_1} \sum_{n_2} N(f(n_1, n_2, t) - f(n_1 - d_1, n_2 - d_2, t-1))$$

$$\text{where } N(\alpha, \beta) = \begin{cases} 1, & \text{if } |\alpha - \beta| > T_0 \\ 0, & \text{if } |\alpha - \beta| \leq T_0 \end{cases}$$

The absolute difference operator inside the above function can be replaced by the squared difference operator or any other appropriate threshold detector.




20




■ **Remarks:**

1. To estimate the motion vector, we normally maximize the value of NCF or minimize the values of the other three functions.
2. In detection theory, if the total noise, a combination of coding error and the other factors violating our motion assumptions, can be modeled as white Gaussian, then the NCF is the optimal matching criterion. However, the white Gaussian assumption is not completely valid for real images. In addition, the computation requirement of NCF is enormous. \Rightarrow The other matching functions are regarded as more practical, and they perform almost equally well for real images.
3. NTD can be adjusted to match the subjective thresholding characteristics of the human visual system.
4. MAD is the most popular choice in designing practical image coding systems because of its good performance and relatively simple hardware structure.




21



■ **Fast Search Algorithms:**

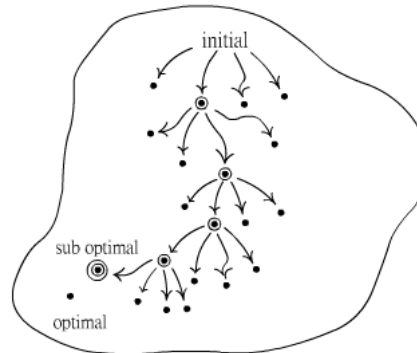
■ **Basic principle:**

- Breaking up the search process into a few sequential steps and choosing the next-step search direction based on the current-step result.
- At each step, only a small number of search points are calculated. Therefore, the total number of search points is significantly reduced.
- Because the steps are performed in sequential order, an incorrect initial search direction may lead to a less favorable result. Also, the sequential search order poses a constraint on the available parallel processing structure.

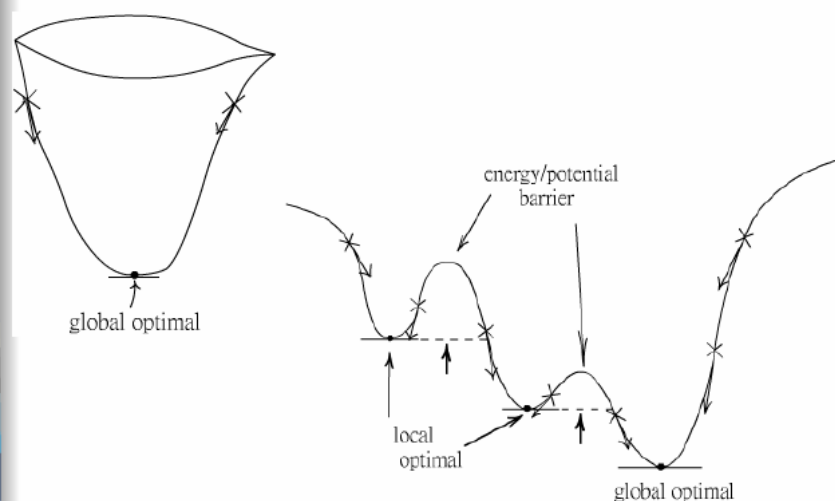


22

Normally, a fast search algorithm starts with a rough search, computing a set of scattered search points. The distance between two nearby search points is called (search) step size. After the current step is completed, it then moves to the most promising search points and does another search with probably a smaller step size.




23




Simulated Annealing
Genetic Algorithm
Neural Networks,
Support Vector Machine




24



If the matching function is “monotonic” along any direction away from the optimal point, a well-designed fast algorithm can then be guaranteed to converge to the global optimal point. But in reality the image signal is not a simple Markov process and it contains coding and measurement noises; therefore, the monotonic matching function assumption is often not valid and consequently fast search algorithms are often suboptimal.



25

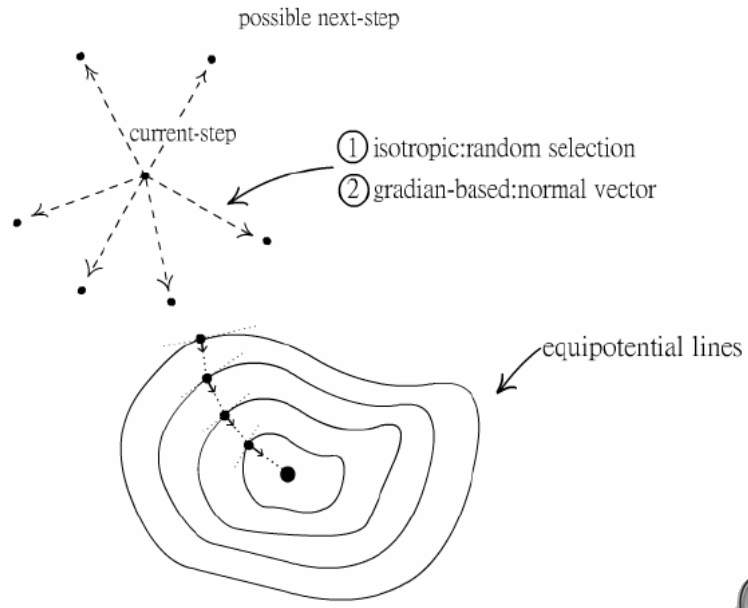



possible next-step

current-step

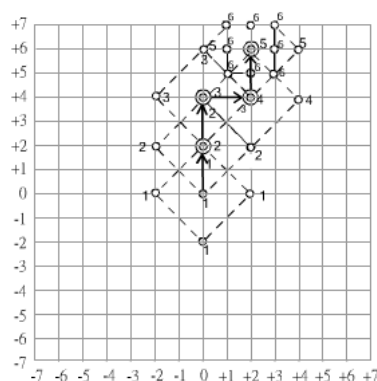
① isotropic: random selection
② gradient-based: normal vector

equipotential lines

26

2-D-log search procedure

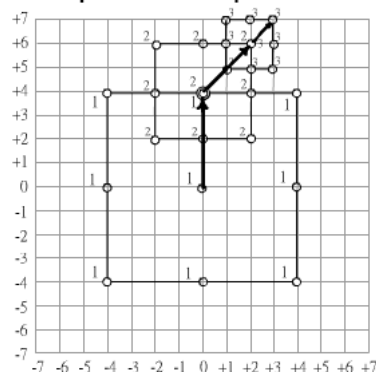


- Diamond-shaped search area(at most 5-point/step)
- 9 search points in 3X3 area surrounding the last best matching point are compared.
- The step size is reduced to half of its current value if the best match is located at the center or located on the border of the maximum search region.



27

3-step search procedure



- The search starts with a step size equal to or slightly larger than half of the max search range.
- 9-points are compared in each step.
- The step size is reduced by half, after each step, and the search ends with step size of 1 pel.



28

■Remarks:

1. A threshold function is used to terminate the search process without reaching the final step. - As long as the matching error is less than a small threshold, the resultant motion vector would be acceptable.
2. One-at-a-time search:
Separate a 2-D search problem into two 1-D problems: looks for the best matching point in one direction first, and then looks in the other direction.



29

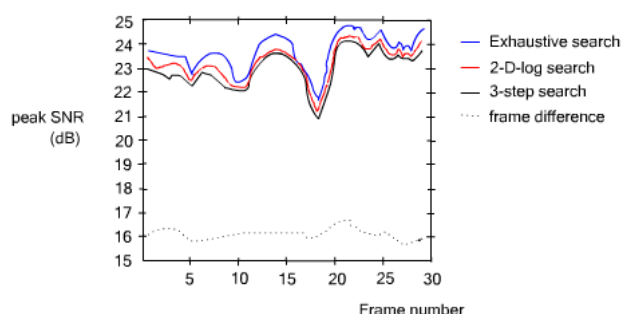
3.

Search algorithm	Number of search points		Number of search steps	
	min	max	min	max
Exhaustive search	225	225	1	1
2D-log search	13	26	2	8
3-step search	25	25	3	3
Modified-log search	13	19	3	6
One-at-a-time	5	7	2	14
Orthogonal search	13	13	6	6



30

4. Performance



5. In hardware systems, the exhaustive search and the three step search are often favored for their good PSNR performance, their fixed and fewer number of search steps, and their identical operation in every step.



31

■ Variants of Block Matching Algorithms

The computation load could also be reduced by calculating fewer blocks of an image.

- To increase search efficiency, we could place the initial search point at a location predicted from the motion vectors of the spatially or the temporally adjacent blocks. A best matching can often be obtained by searching a smaller region surrounding this initial point.
- We could first separate the moving image blocks from the stationary ones and then conduct block matching only on the moving objects. This is because a moving or change detector can be implemented with much fewer calculations than a motion estimator.
- We could use only a portion of the pels inside an image block (subsampling images) in calculating the matching function.
→ subsampling pattern v.s. accuracy of motion
- Perform the motion estimation only on the alternate blocks in an image; the motion vectors of the missing blocks are “interpolated” from the calculated motion vectors.



32

small block size \Rightarrow ambiguity noise problem
 large block size \Rightarrow inaccuracy problem

\Rightarrow Hierarchical block matching
 Variable-block-size approach

■ Basic principle:

A large block size is chosen at the beginning to obtain a rough estimate of the motion vector. Because a large-size image pattern is used in matching, the ambiguity problem-blocks of similar content-can often be eliminated. However, motion vectors estimated from large blocks are not accurate. We then refine the estimated motion vectors by decreasing the block size and the search region.

A new search with a smaller block size starts from an initial motion vector that is the best matched motion vector in the previous stage. Because pels in a small block are more likely to share the same motion vector, the reduction of block size typically increases the motion vector accuracy.

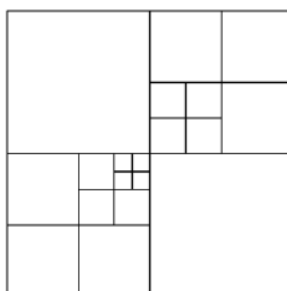


33

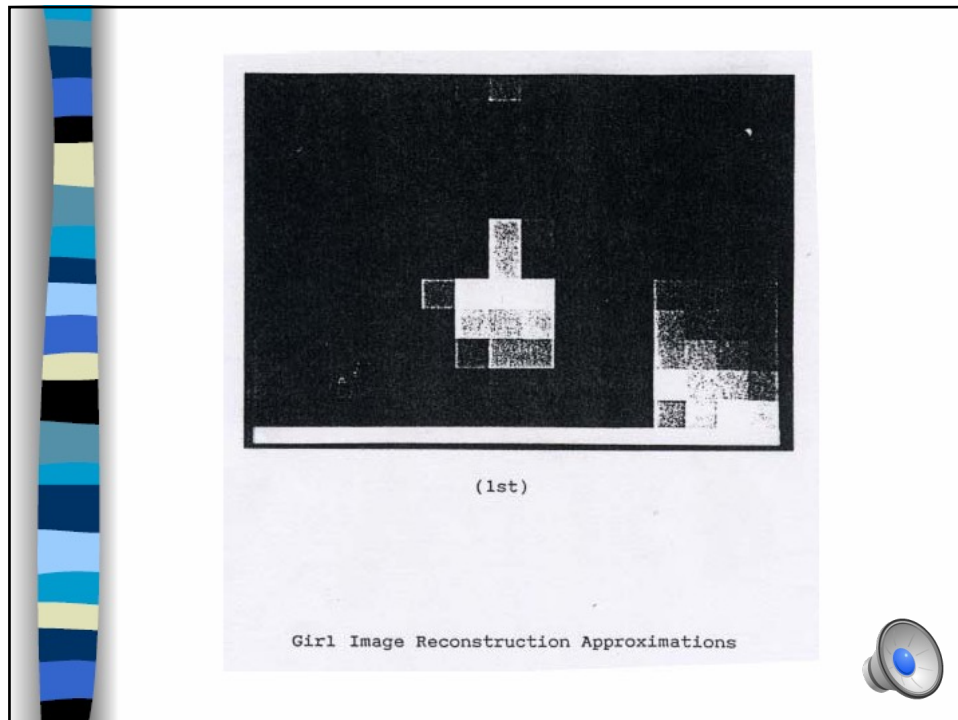
■ Variable-block-size motion estimation:

Image frames are partitioned into non-overlapped large image blocks. If the motion-compensated estimation error is higher than a threshold, this large block is not well-compensated; therefore, it is further partitioned into, say, four smaller blocks.

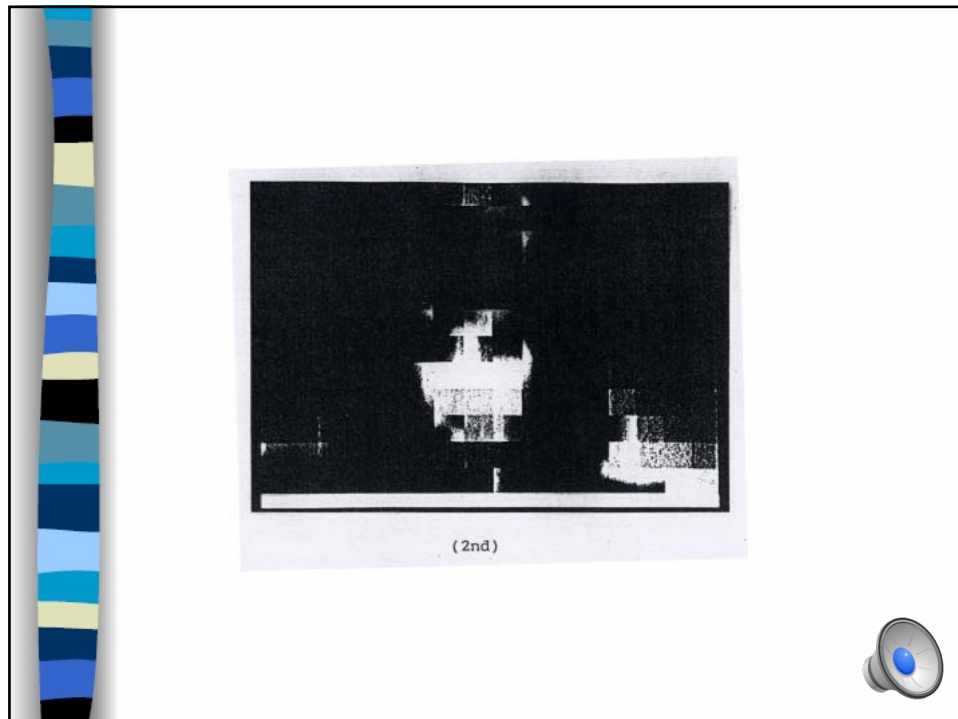
In searching for the motion vector is used as the initial search location.



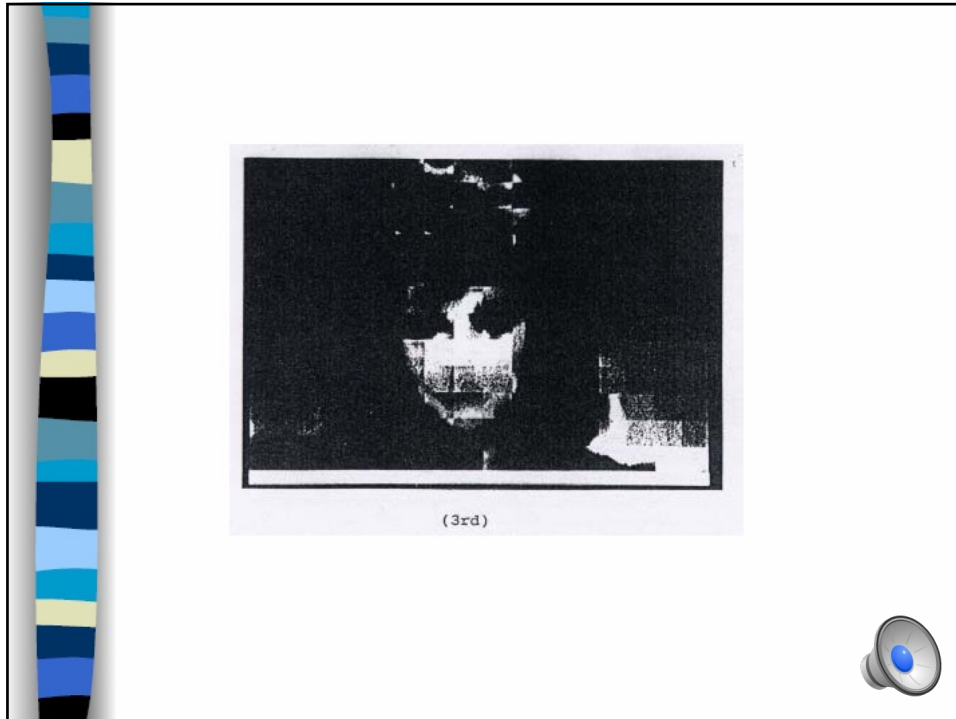
34



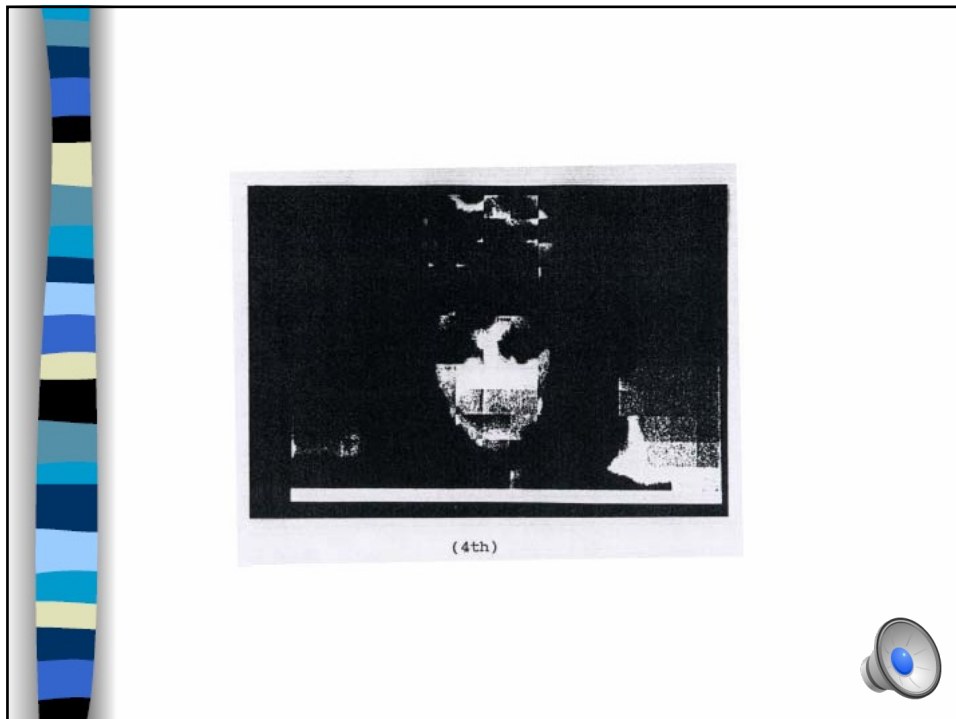
35



36



37



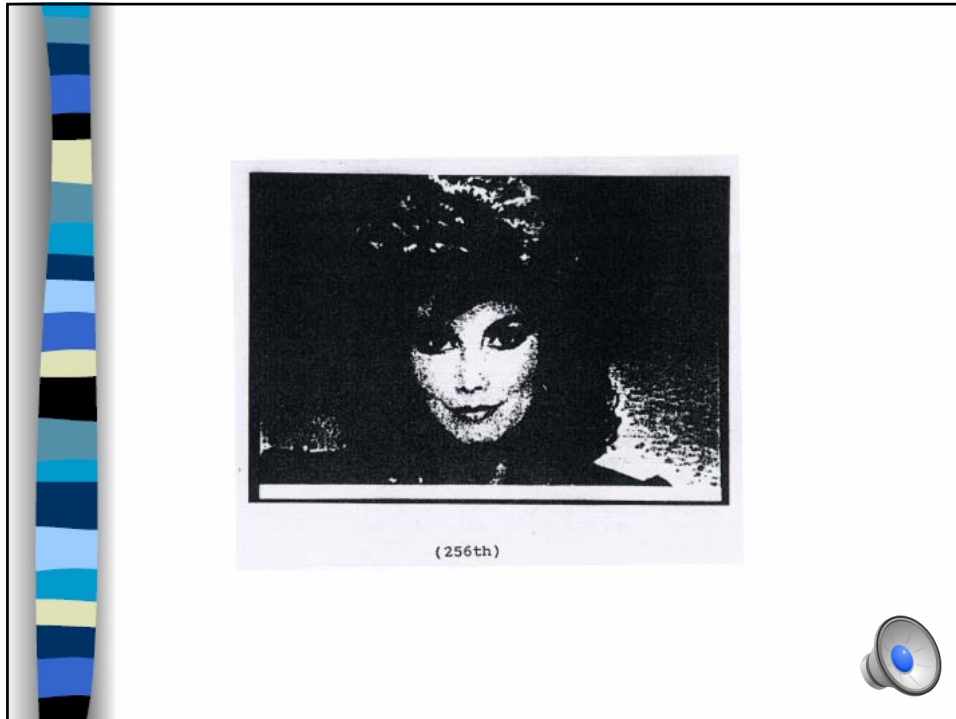
38



39



40



41



42