Stereo Image Compression

Group 1

Charles Dunn, Oren Hazi, & Marland Sitt

Overview

- Introduction
- Methods
- Results
- Additional ideas
- Challenges
- Questions

Motivation

- Stereo images pairs
  - Perspective correlation
  - Highly compressible
- Conventional image encoding does not take advantage of perspective correlation
- Ideally code two images at a small marginal cost to coding a single image

Problem Definition

- Stereo images
  - Y, Cb, Cr format (Cb, Cr ↓2)
  - uint8 format
  - 960x540 stereo images
- 14 stereo images provided
  - very different disparities
  - very different frequency components
- 37 dB minimum PSNR
- 14 min to process 7 images
- Highest compression ratio (CR)

Algorithm Overview

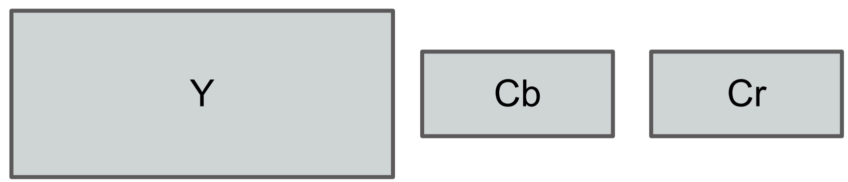
- Base image encoding
- Disparity vector estimation
- Disparity vector encoding
- Subpel disparity vectors
- Residual encoding
- Decoding

Base Image Encoding

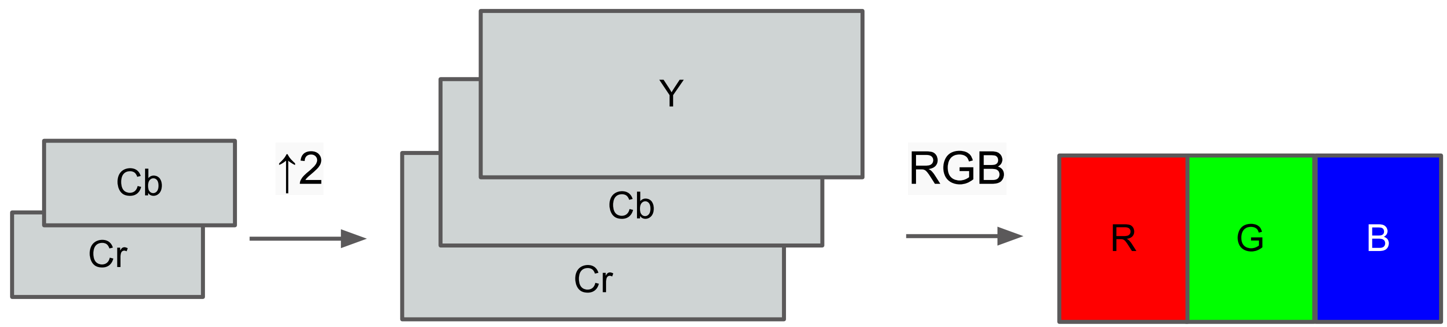
- Left image chosen
- JPEG-2000
  - Focused on exploiting cross-image correlation
  - Unconvinced that we could beat JPEG-2000
- Compression arrangements
  - Separate
  - Combined
  - Patched

Compression Arrangements

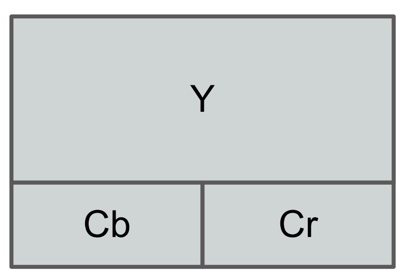
Separate



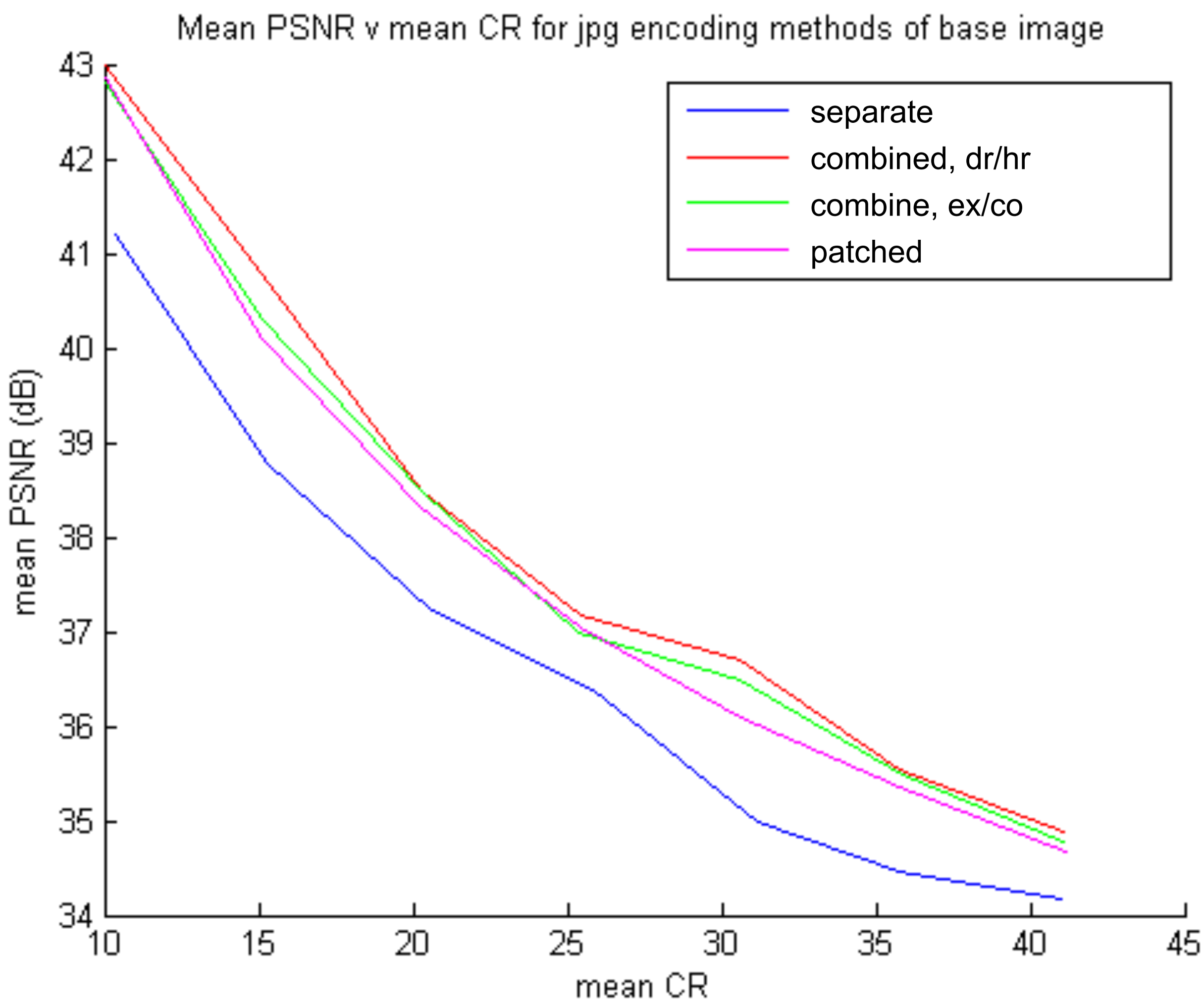
Combined



Patched



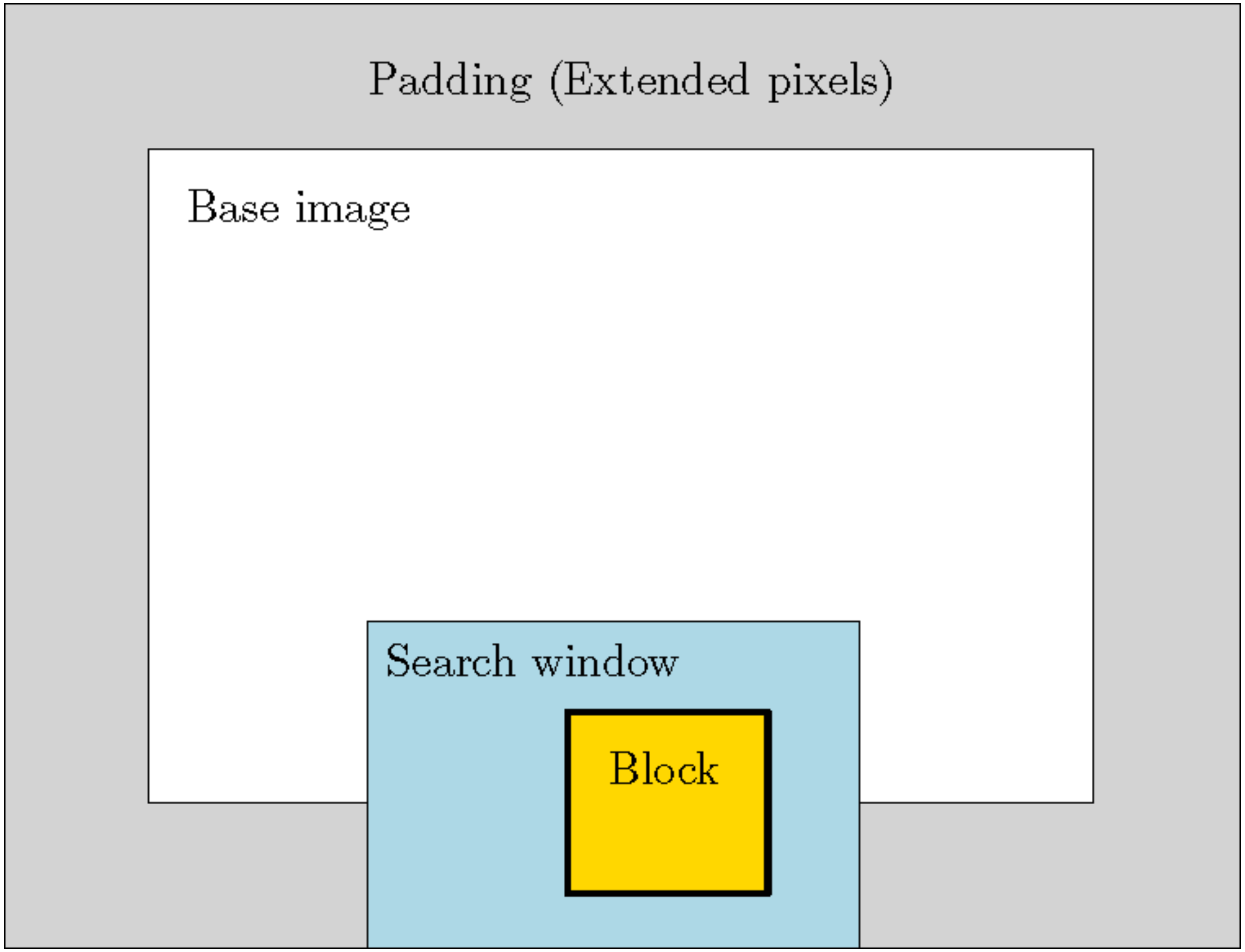
Compression Arrangements



Disparity Vector (DV) Estimation

- Ideally, only lie along epipolar line
- Motivated by video motion vector search
- Block by block estimation over a limited search window
- DV choice affects
  - Bits needed to code disparity vectors
  - Bits needed to code residuals
- MSE cost function for simplicity
- Quarter-pel accuracy

Disparity Vector Search Algorithm



Search Algorithm Performance

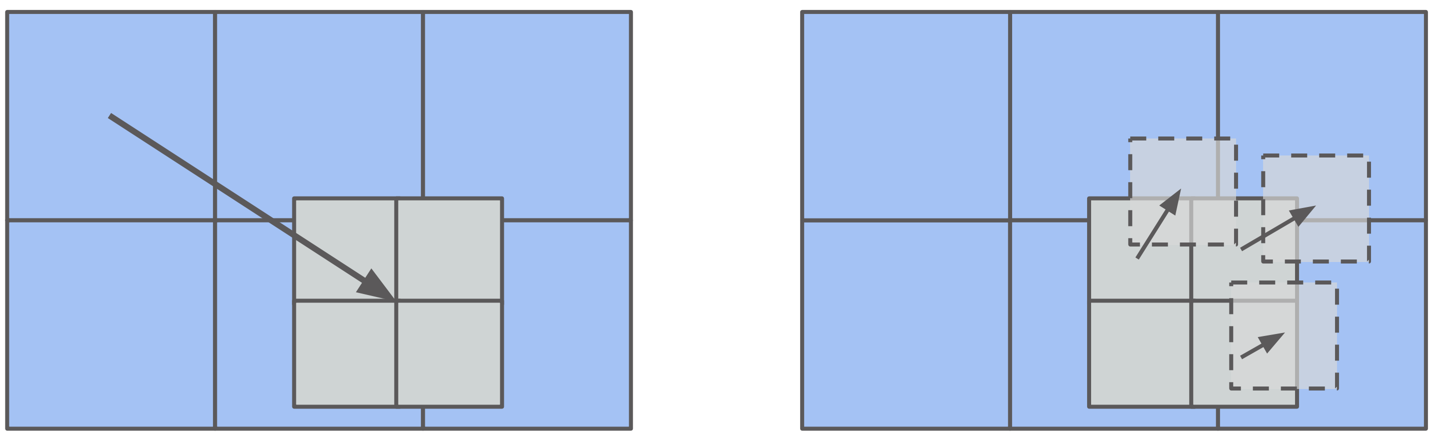
- Early termination improves performance
- Overhead from spiral search too high

(N, window size)	Naive approach	Early termination	ET, spiral search
(8, 20)	0.59	0.50	0.59
(8, 40)	1.17	1.00	1.19
(8, 50)	1.51	1.31	1.52
(16, 20)	0.42	0.31	0.42
(16, 40)	1.10	0.83	1.08
(16, 50)	1.43	1.13	1.43

- All values are in seconds

DV Estimation Algorithms

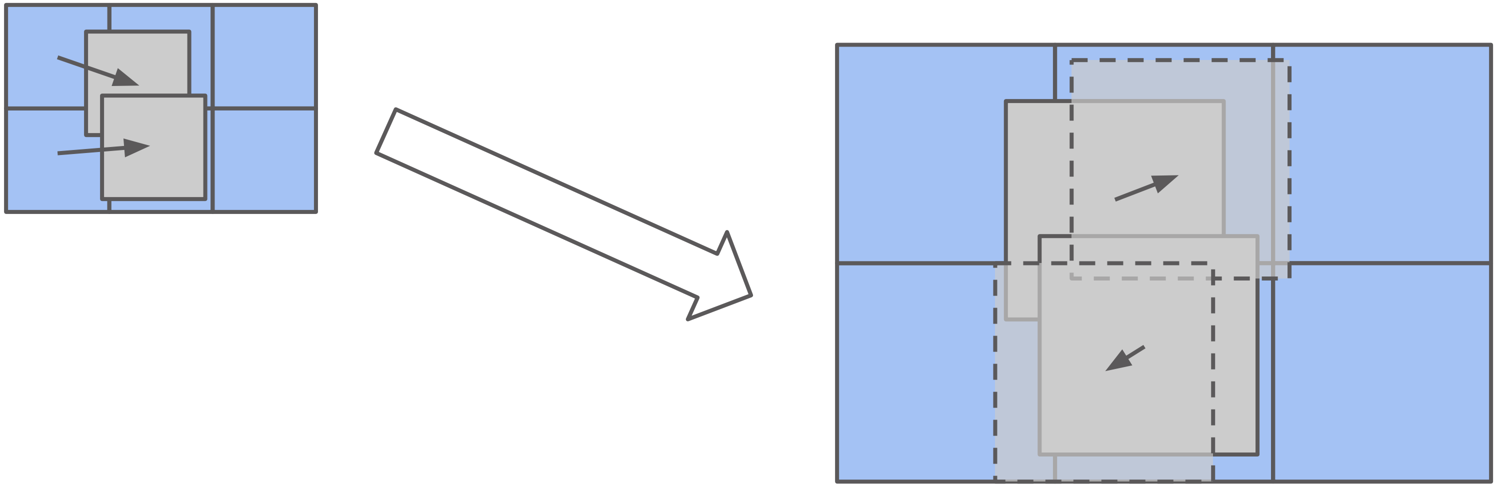
- Top Down Search



- Top-down search forces a smoothly varying disparity vector field

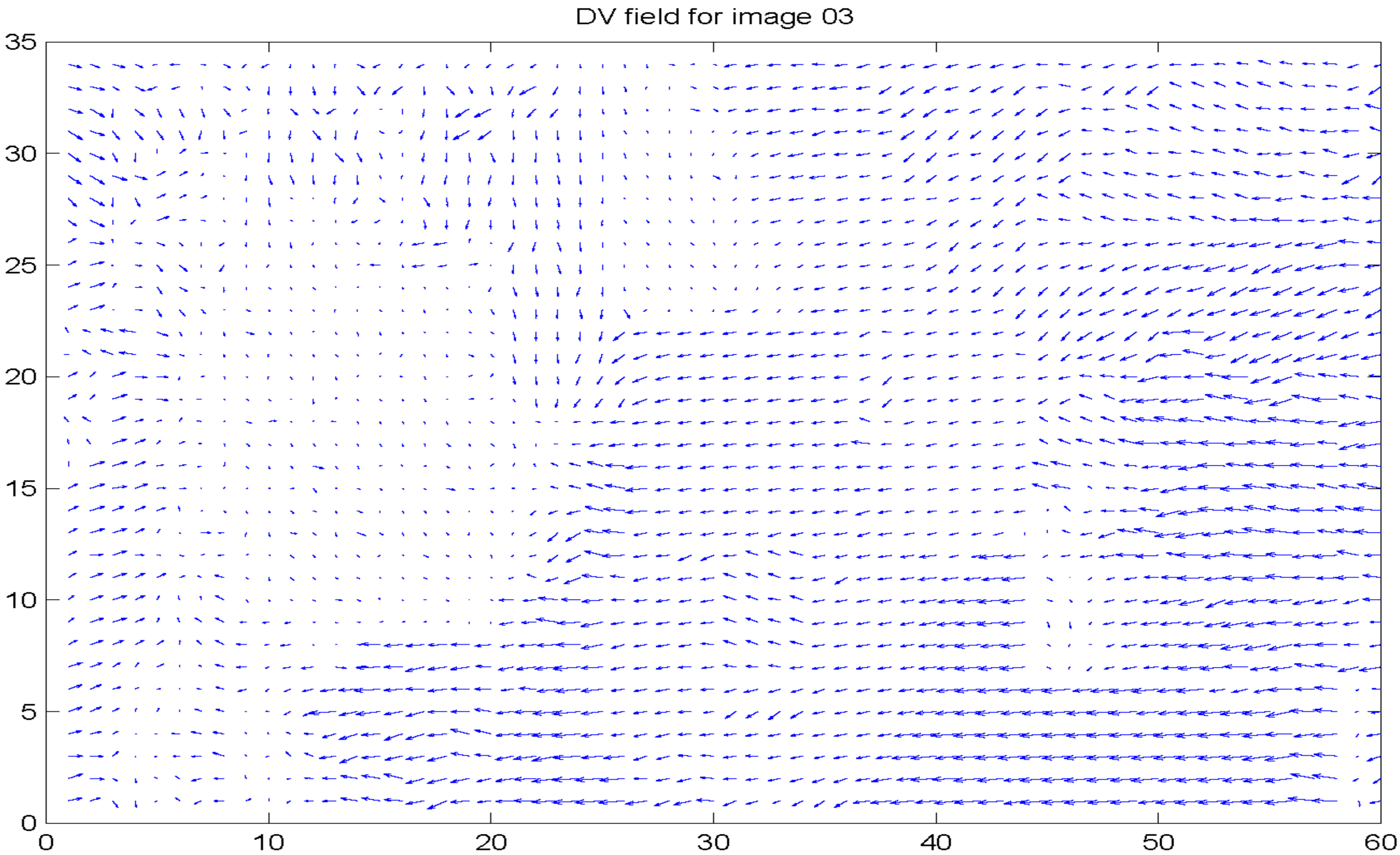
DV Estimation Algorithms (2)

- Scale search

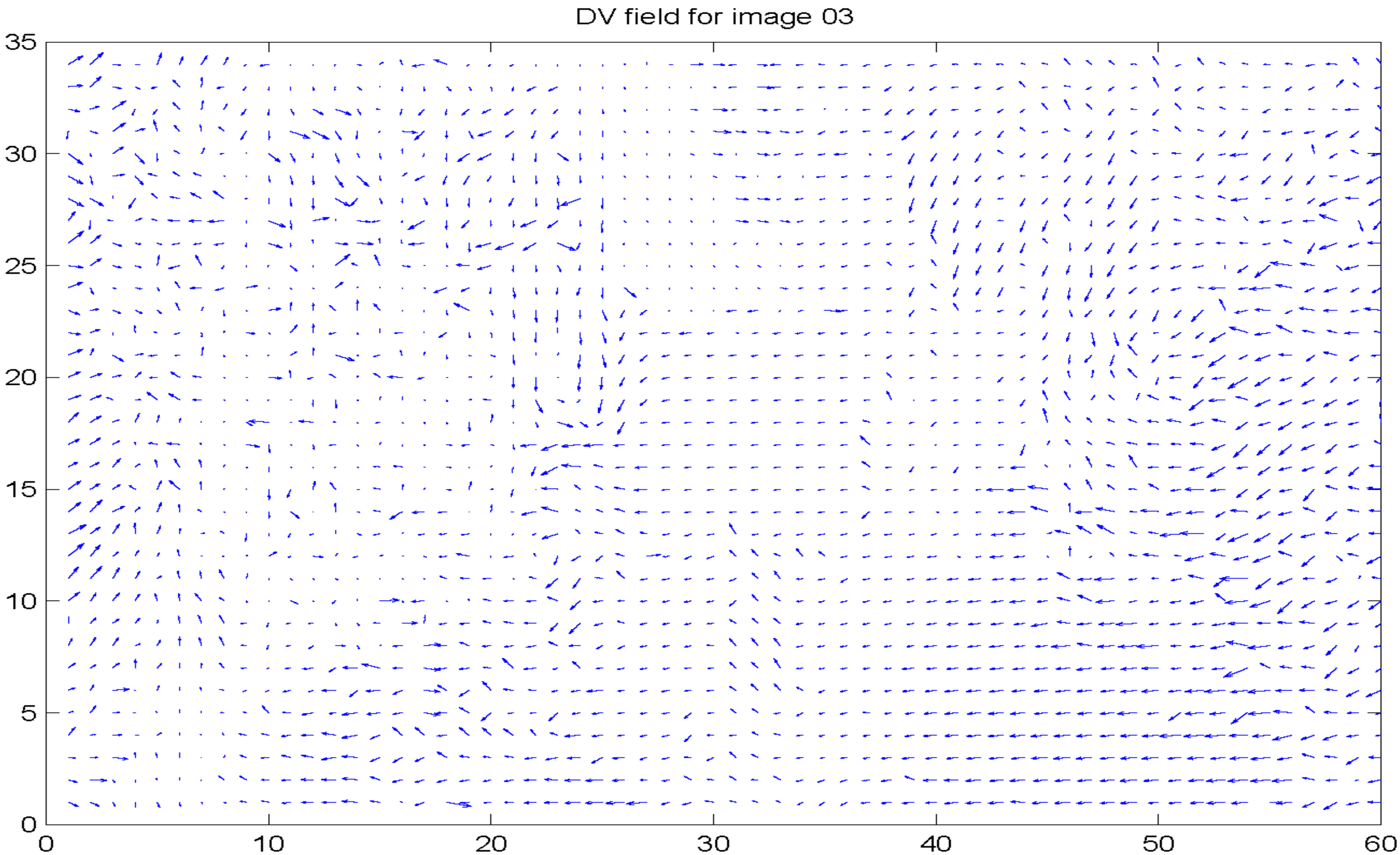


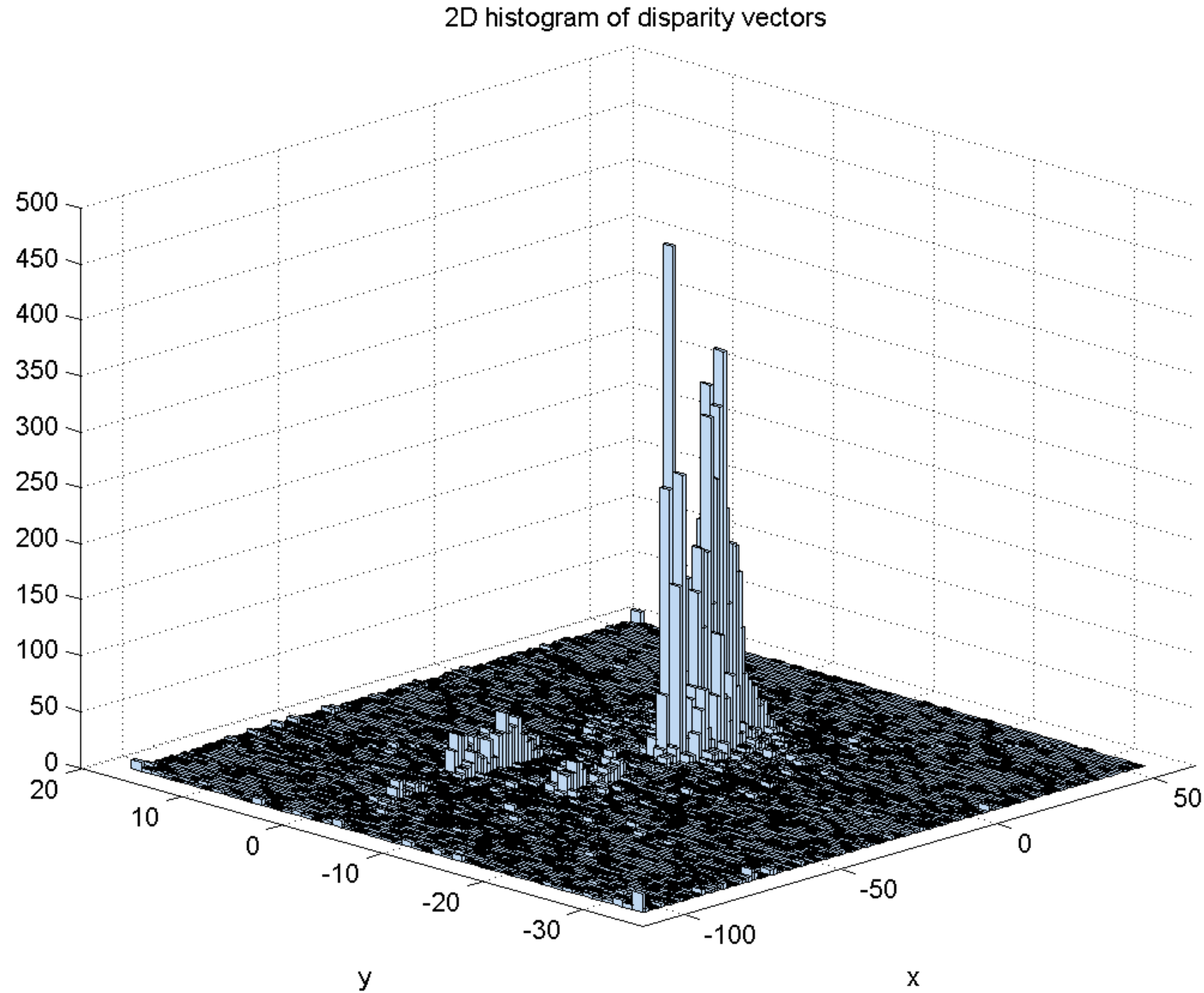
- Allows for wide search window by searching over discrete grid

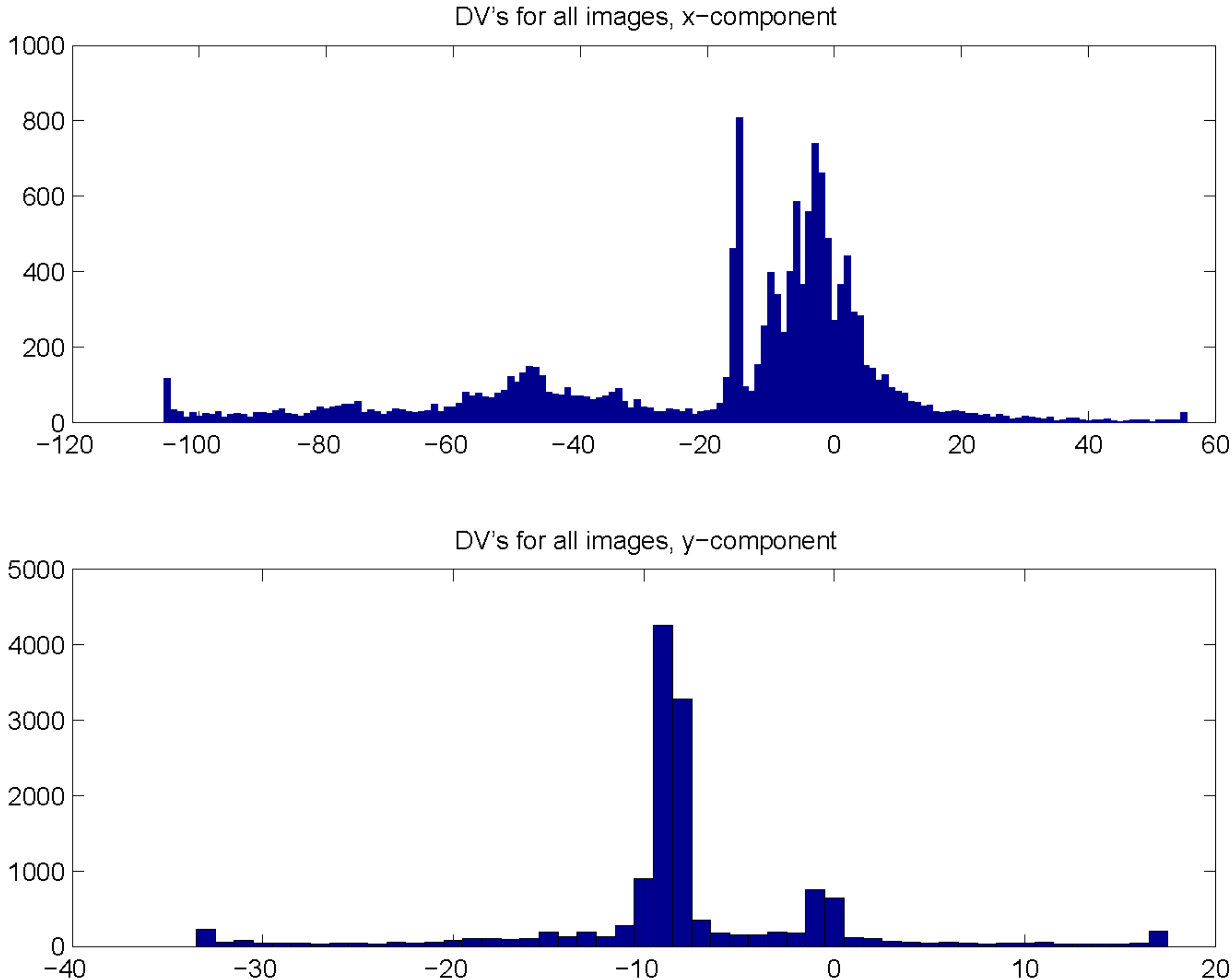
Top-down search

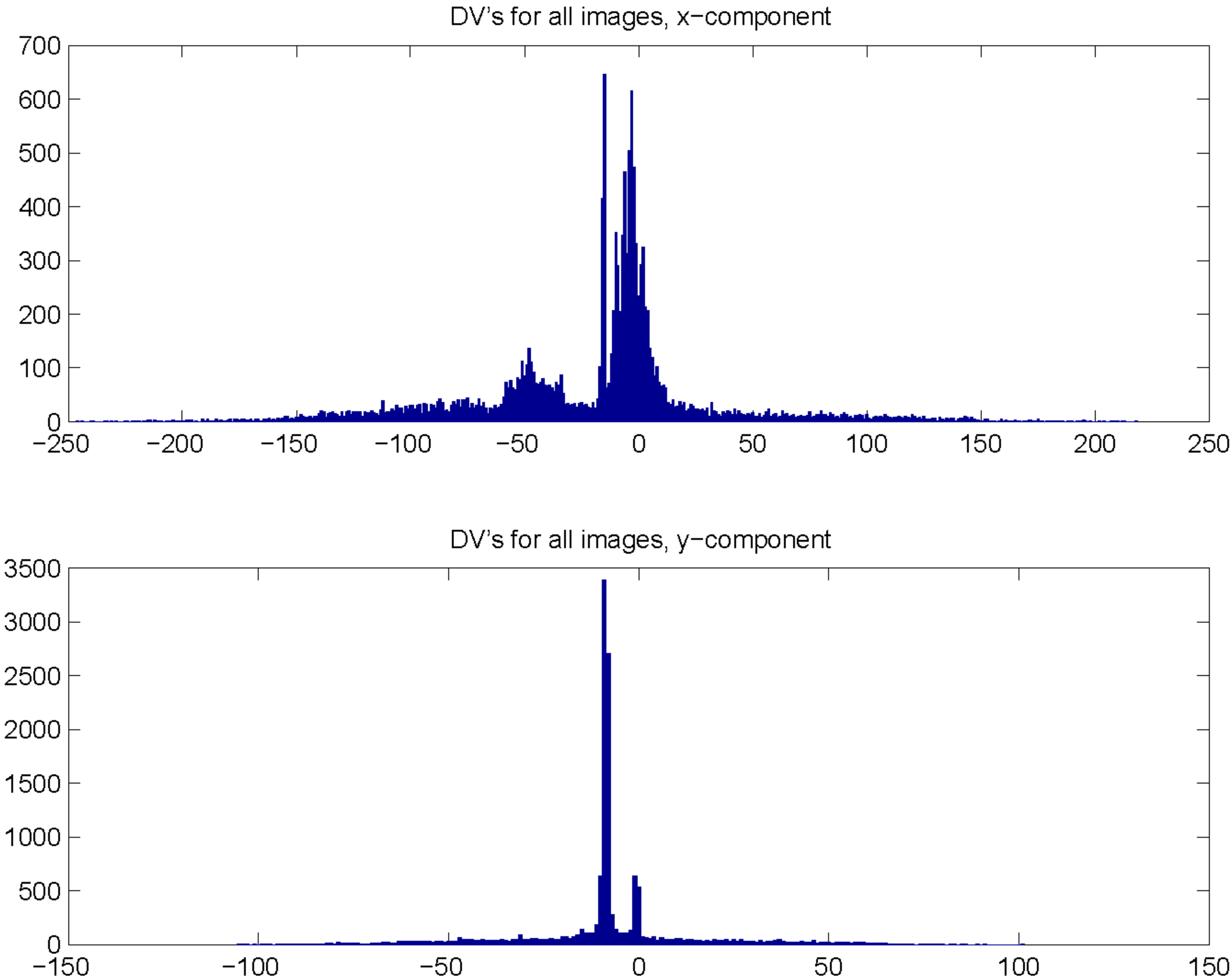


Scale search



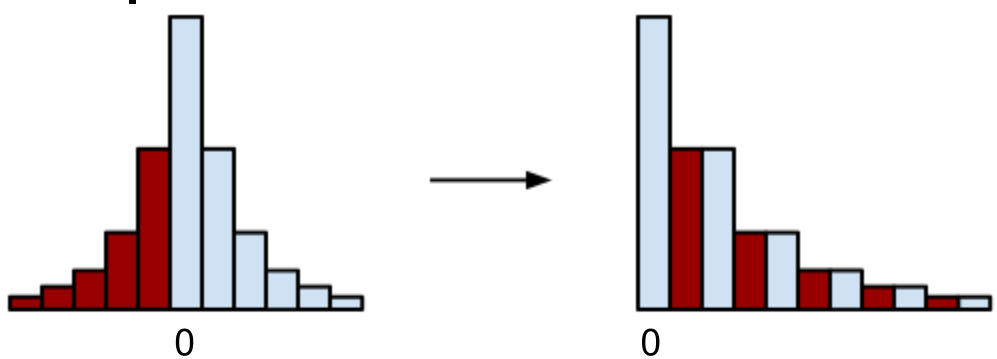

2D histogram of disparity vectors
500
450
400
350
300
250
200
150
100
50
0
20
10
0
-10
-20
-30
y
-100
-50
0
50
x


DV's for all images, x−component
1000
800
600
400
200
0
−120
−100
−80
−60
−40
−20
0
20
40
60
DV's for all images, y−component
5000
4000
3000
2000
1000
0
−40
−30
−20
−10
0
10
20


DV's for all images, x−component
700
600
500
400
300
200
100
0
−250
−200
−150
−100
−50
0
50
100
150
200
250
DV's for all images, y−component
3500
3000
2500
2000
1500
1000
500
0
−150
−100
−50
0
50
100
150

Disparity Vector Coding

- X and Y DVs coded separately
- Zig-zag differential coding
- Map to all positive values

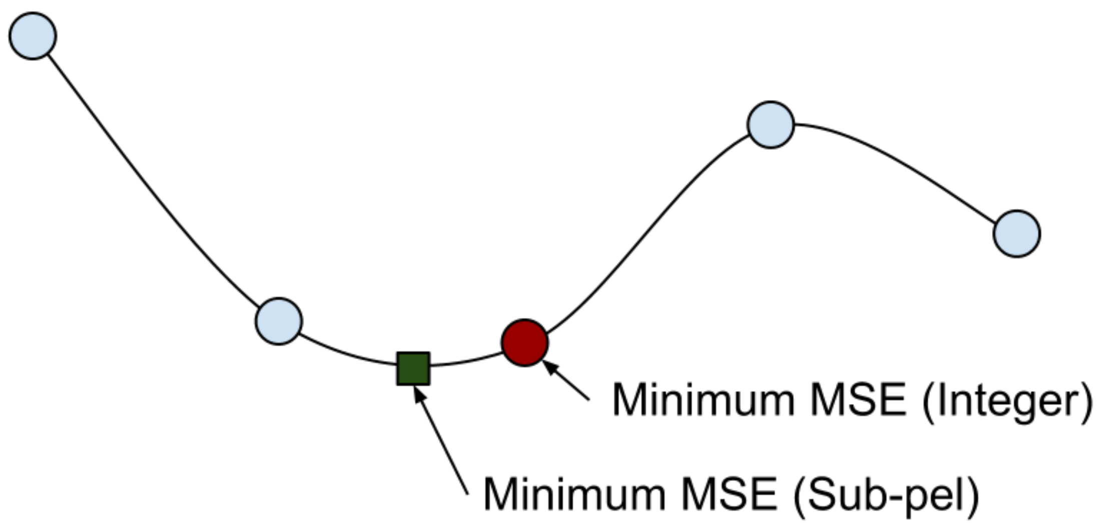


Golomb coder

- Compression ratio varied from 1.25 - 2.2
    - CR for y component was usually higher by ~ 15%

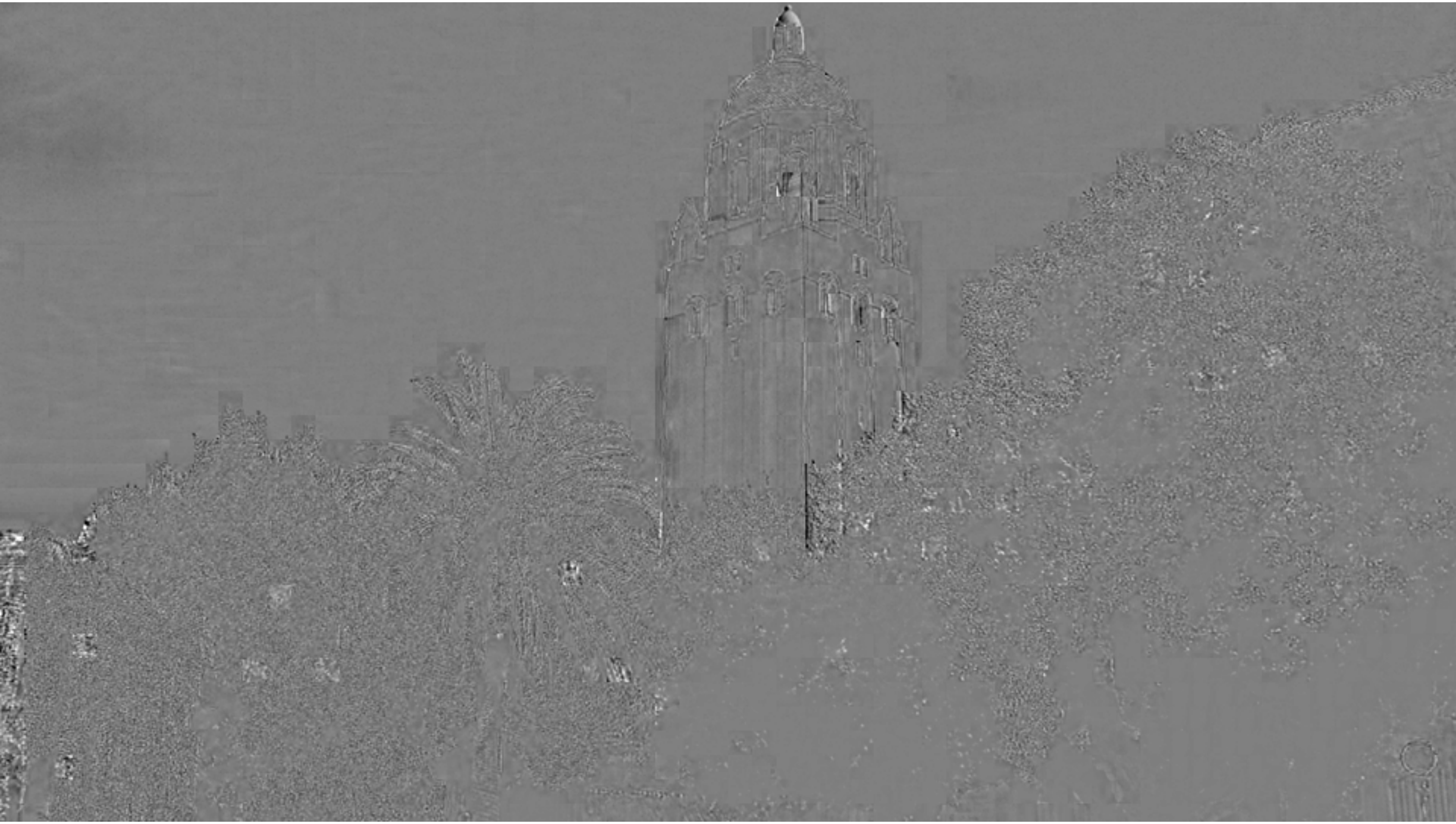
Quarter-Pel Disparity Estimation

- Compute MSE values for integer block shifts
  - Bicubic interpolation for subpel DV
  - Only interpolate 5x5 region around global minimum integer MSE value



Residuals using Integer-Pel Disparity Vectors

Residuals using Quarter-Pel Disparity Vectors



Residual Encoding

- Two methods of residual encoding
  - DCT followed by Huffman coding
  - JPEG-2000 encoding
- Method resulting in higher CR was selected

Residual Quantization

- Discrete Cosine Transform (DCT)
- Determine target PSNR for second image
- PSNR can be calculated directly from residuals in DCT domain
- Binary search over quantization matrices

$$d_R = \sum (R_{rec} - R_{orig})^2$$

$$R_{rec} = r_Q + L_{rec}, \; r = R_{orig} - L_{rec}$$

$$d_R = \sum ((r_Q + L_{rec}) - (r - L_{rec}))^2 = \sum (r_Q - r)^2 = d_{res}$$

Huffman Residual Encoding

- Difference coding on DC components (zigzag order)
- Run level coding on AC components
- Huffman coding
- Send optimal tables
  - Rather than store suboptimal tables
- Tables compressed to bit strings

Decoding

- Determine encoding method
- Decode base image
- Decode disparity vectors
- Decode residuals
- Reconstruct secondary image

Results

Image	JPEG-2000	Huffman Residual Coding	JP2 Residual Coding
1	10.0350	8.8253	**10.0873**
2	32.4873	**36.6317**	**37.8367**
3	195.3523	145.987	154.1481
4	42.2930	**52.0465**	**54.4423**
5	26.4931	**36.3025**	**37.8624**
6	16.8792	**19.8659**	**20.1422**
7	9.7854	**10.3284**	**10.6564**

- Bold entries indicate compression ratio better than JPEG-2000
- Highlighted entries indicate best performance for a given image

Results (cont.)

Image	JPEG-2000	Huffman Residual Coding	JP2 Residual Coding
8	11.4483	**14.6274**	**14.7235**
9	44.2384	**55.5488**	**59.1713**
10	10.4480	**13.5254**	**13.3902**
11	25.5596	**33.0044**	**34.9743**
12	52.2791	**61.9380**	**70.7295**
13	7.9713	**9.3048**	**9.3956**
14	7.2194	6.2316	6.2801

- Bold entries indicate compression ratio better than JPEG-2000
- Highlighted entries indicate best performance for a given image

Processing Time

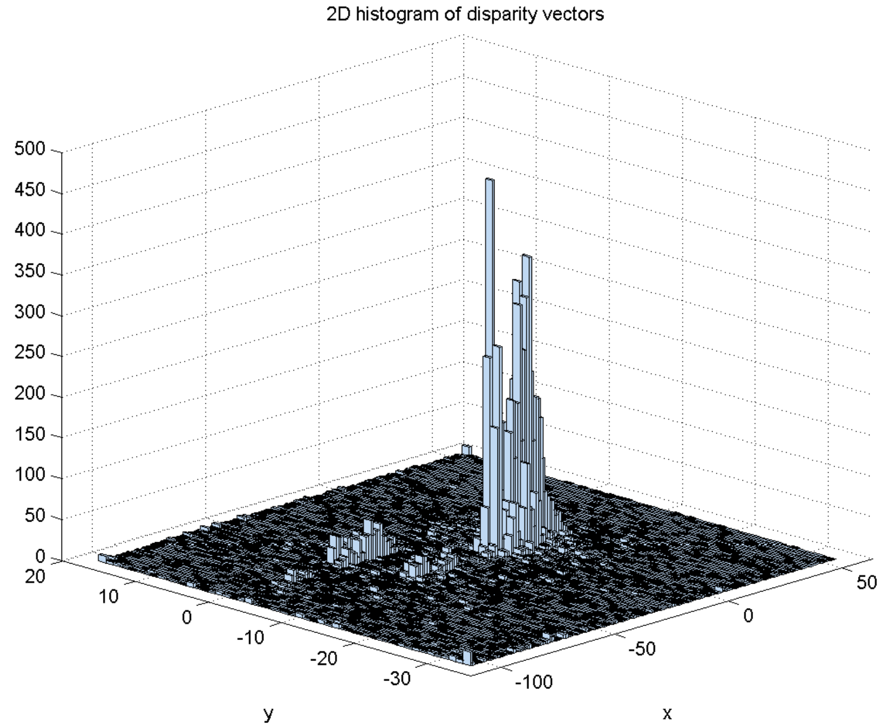
Task	Processing time (seconds)
Base image encoding	**3**
Disparity vector search	**36**
Residual quantization	**8**
Residual Huffman coding	**3**
Golomb coding	**1**
Write to disk	**1**
Decoding	**6**
Total	**58**

Additional Ideas

- Different PSNRs for base image
- Sub-zigzag order of DVs
- Optimization of quantization tables
- Variable block size for DV estimation
- Copy mode
- 1/8 pel DVs for Cb and Cr

Challenges

- Time constraint
  - Many optimizations necessary
  - Suboptimal algorithms
- Different image sources in training set



Questions