Matching quality investigation of Motion Estimation Algorithms

Alexander Setkov
Intel Corp. Russia
196247, St-Petersburg, 160 Leninsky pr., Russia
alex.setkov@intel.com

FRUCT 9th conference - Petrozavodsk

Agenda

- Classification of Motion Estimation algorithms
- Fast FFT-based algorithm
- Experiments on quality of matching
- Results

The main goal is to choose a fast Motion Estimation algorithm for further integration to existing video codecs

Classification of Motion Estimation algorithms

Full Search Algorithms

+ guarantee the best matched candidate
+simple implementation and good parallelization
- Big amount of computations

Fast Algorithms

•With early elimination of candidates
+ decrease computational complexity
-do not guarantee the best matched candidate
-Sometimes they have unpredictable time and size costs

Metrics:

$$SAD(x,y) = \sum_{l=0}^{n-1} \sum_{k=0}^{m-1} |f(x+k, y+l) - g(k,l)|$$

$$SSD(x,y) = \sum_{l=0}^{n-1} \sum_{k=0}^{m-1} (f(x+k, y+l) - g(k,l))^2$$

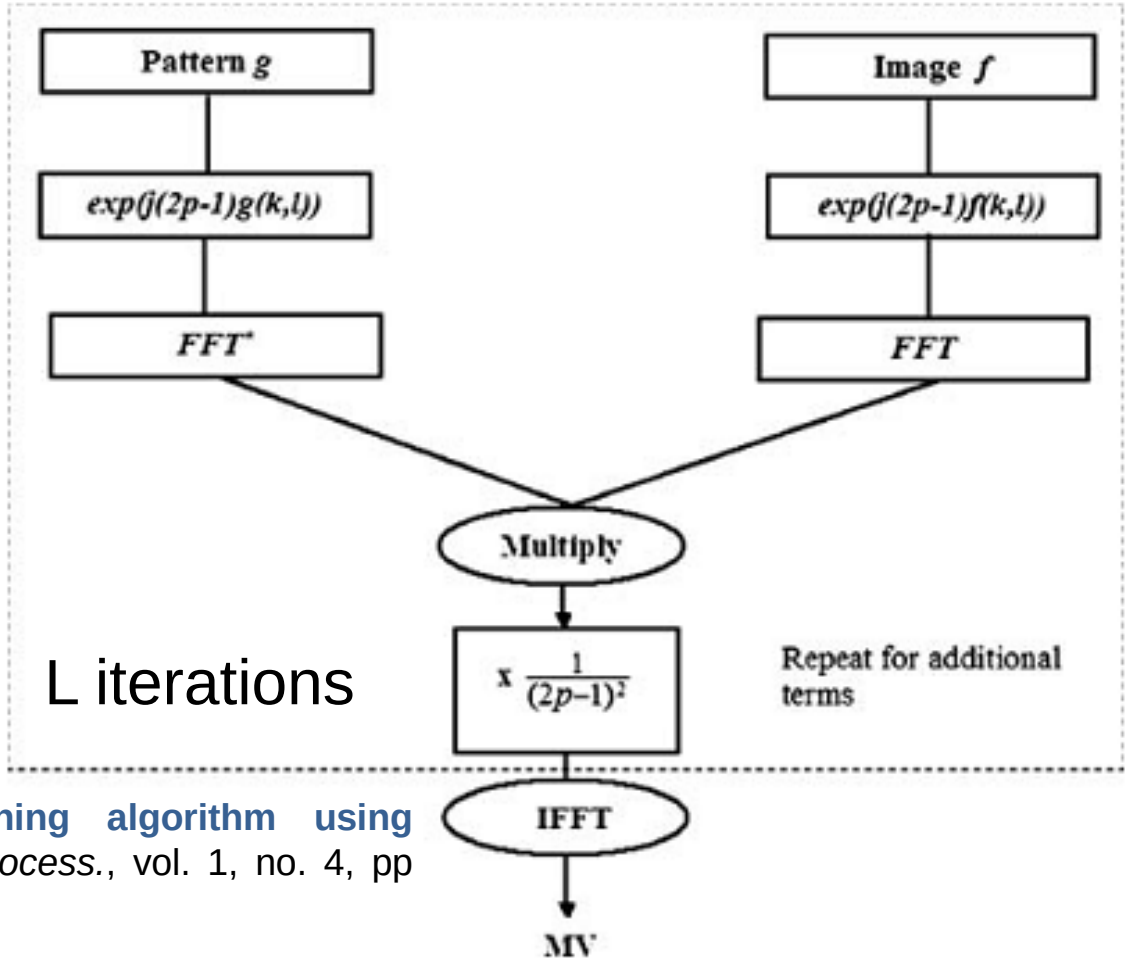
•Calculation in the frequency domain
+ decrease computational complexity
+ predictable time and size costs
+ effective for a hardware implementation (good parallelization)

Fast FFT-based method (1)

makes calculations in the frequency domain. For that FFT is being used as:

- it has constant complexity of n*log(n)
- There are a lot of fast hardware implementations

F – search area in a base frame
g – searched domain
MV – motion vector, maxima corresponds to the best SAD



Essannouni F., **"Adjustable SAD matching algorithm using frequency domain"**, *J. Real-Time Image Process.*, vol. 1, no. 4, pp 105-107, Jan. 2007

Fast FFT-based method (2)

Arithmetic complexity:

$$O = M^2 \cdot (2 + 36 \cdot L + logM \cdot (4L + 2))$$

where: M – search area size
m – domain size
L – number of iterations

Number of operations

Search area /domain size	Full Search SAD, numb ops	FFT-based method, num ops	Speed-up of FFT-based method
64/16	1844017	327680	82.23
64/8	623865	327680	47.48
64/4	178669	327680	-83.4
32/16	221969	75776	65.86
32/8	120025	75776	36.87
32/4	40397	75776	-87.58

Experiments on quality of matching (1)

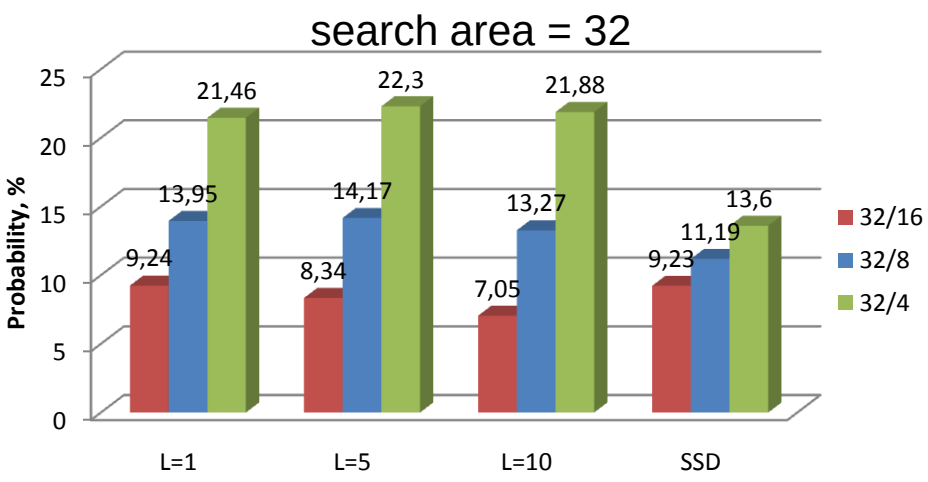
Experiments were carried out on “Carphone” video sequence:

- on first 150 frames with resolution 352x288
- used sizes of search area are 32x32, 48x48, 64x64
- used domain sizes are 4x4, 8x8, 16x16 pels
- used numbers of iterations are 1, 5, 10

Used criteria to estimate the quality:

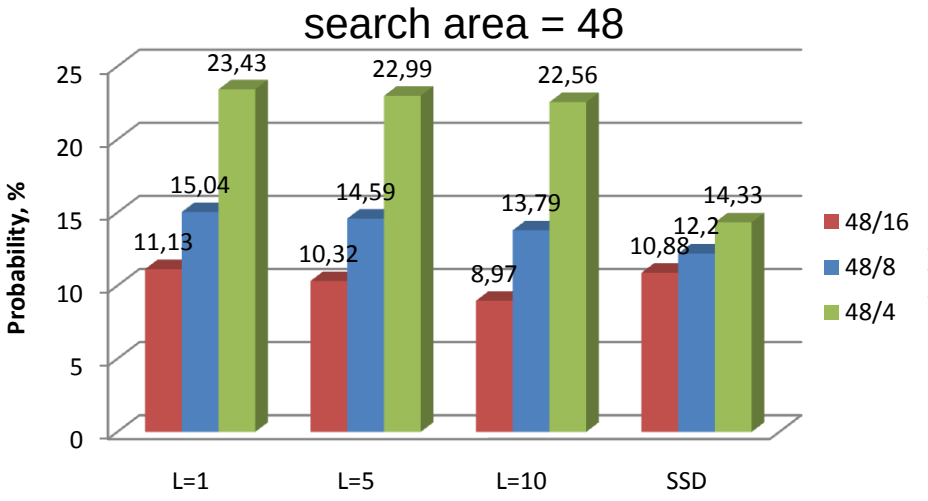
- Error probability of domain matching compared with SAD
- Error probability of domain matching compared with SSD
- PSNR

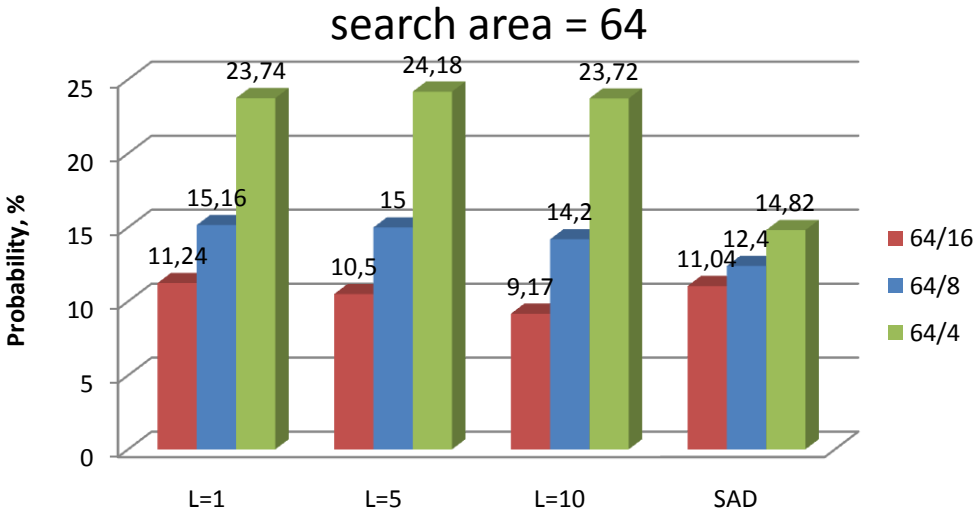
Experiments on quality of matching (2)



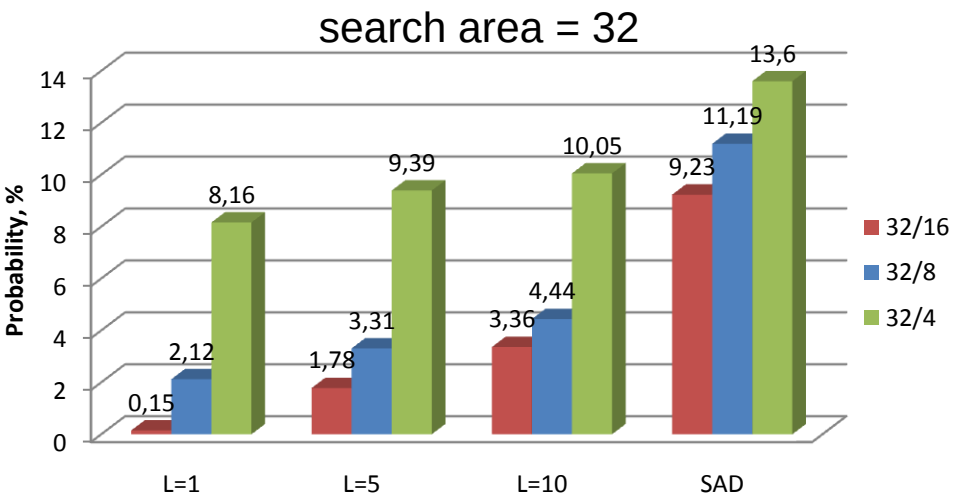
Error probability of domain matching compared with **SAD**

Errors become less with number of iterations increasing
But they remain still significant



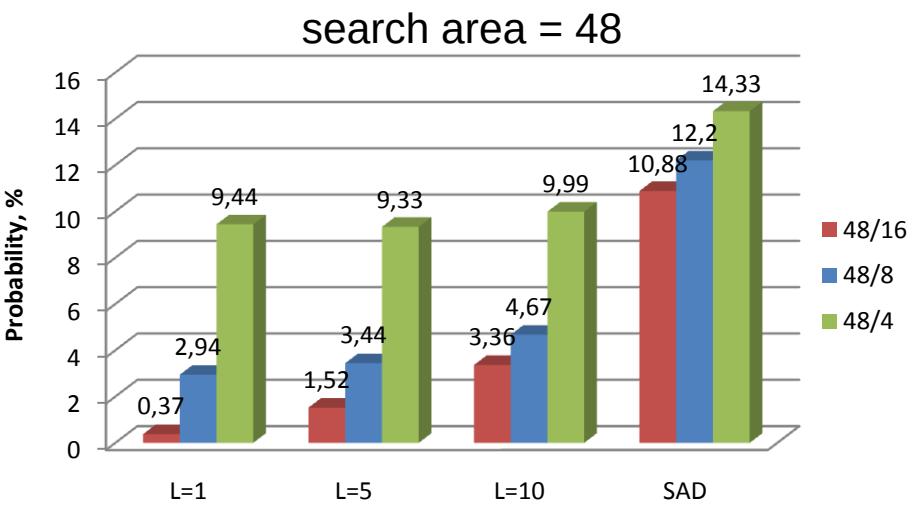


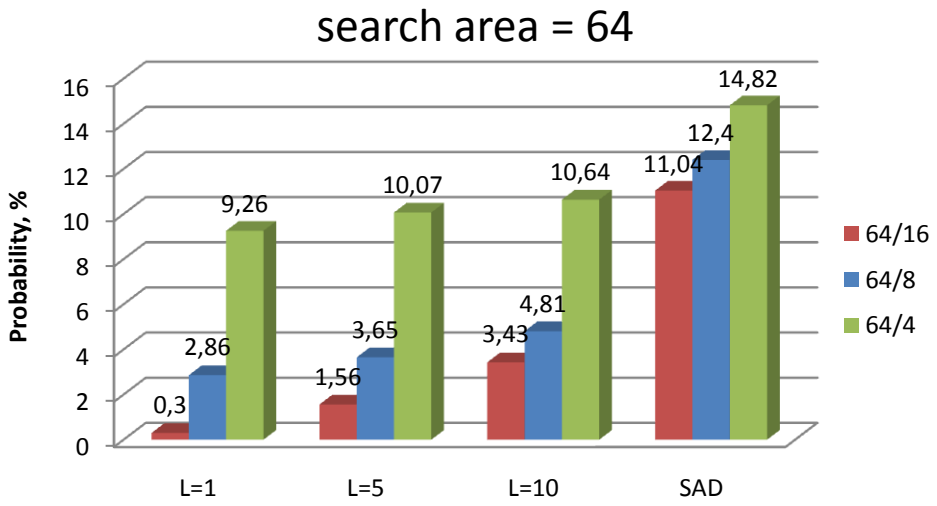
Experiments on quality of matching (3)



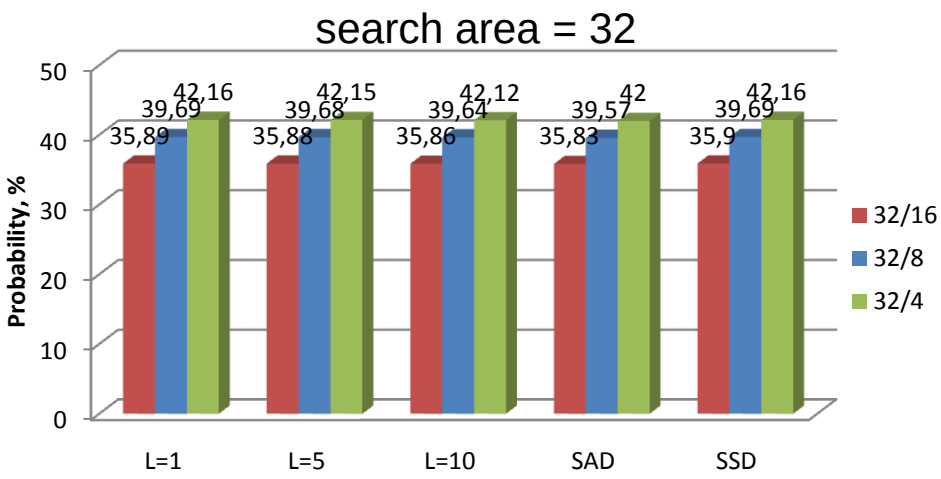
Error probability of domain matching compared with **SSD**

Error probability becomes bigger with number of iterations increasing.
But using only one iteration we get low probability



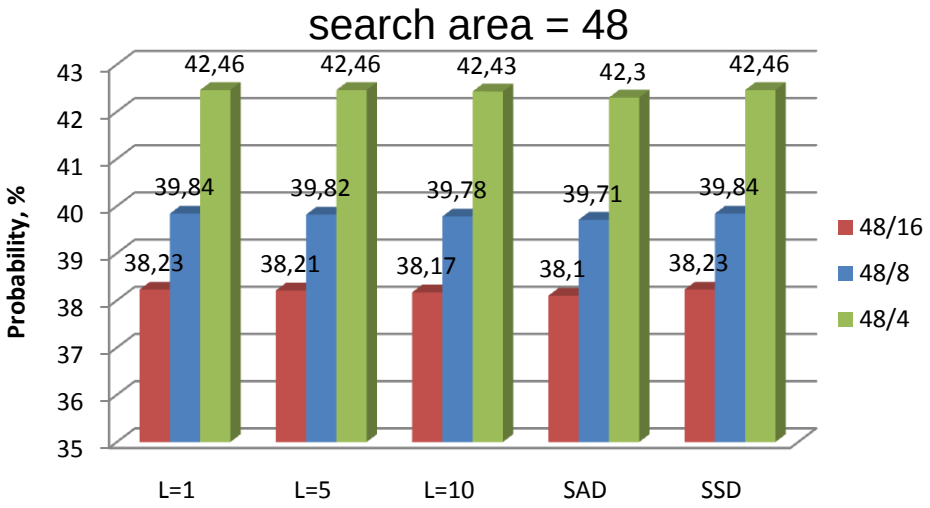


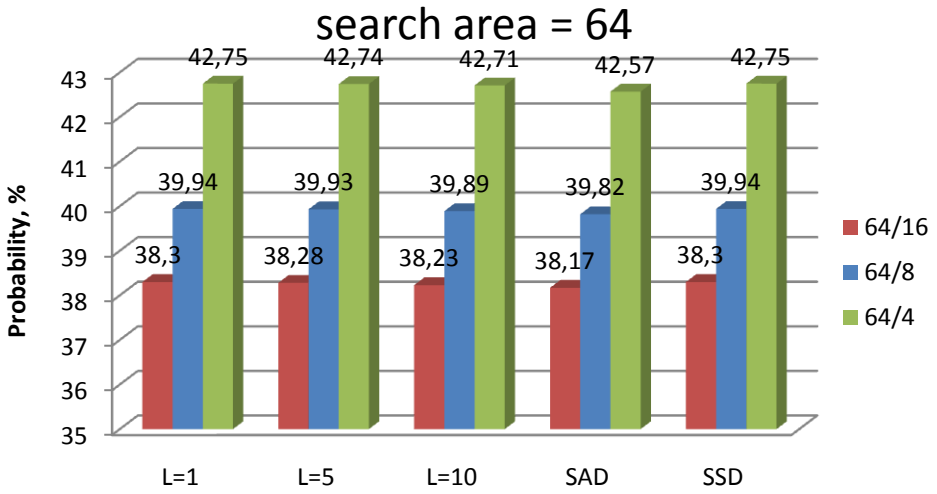
Experiments on quality of matching (4)



PSNR

All PSNR values are closed to each other for all iterations.
It is enough to stop at the first iteration





Results (1)

L = 1

L = 5

Real frame

L = 10

Results (2)

L = 1

L = 5

Real frame

L = 10

Results (3)

Results (4)

We have shown that the FFT-based algorithm:

- gives results closed to SSD Full-Search results
- gives speed-up up to 82% for domain sizes 16x16 pels

This algorithm is suitable for further hardware implementation

Questions ???

email: alex.setkov@intel.com