JOANNEUM RESEARCH DIGITAL

DIGITAL - Institute for Information and Communication Technologies

SIFT Descriptor Extraction on the GPU for Large-Scale Video Analysis

Hannes Fassold, Jakub Rosner

2014-03-26

Overview

- GPU-activities @ AVM research group
- SIFT descriptor extraction
  - Algorithm
  - GPU implementation (Fermi architecture)
- Applications
  - Brand monitoring
  - Near-duplicate Video detection

Overview

- GPU-accelerated algorithms / applications @ AVM
  - AVM .. Audiovisual Media research group, DIGITAL – Institute for Information and Communication Technologies, JOANNEUM RESEARCH
  - Content-based video quality analysis
    - http://vidicert.com
  - Digital film restoration
    - http://www.hs-art.com
  - Brand monitoring
    - http://www.branddetector.at
  - …
  - GPU – related activities since 2007, using CUDA C++

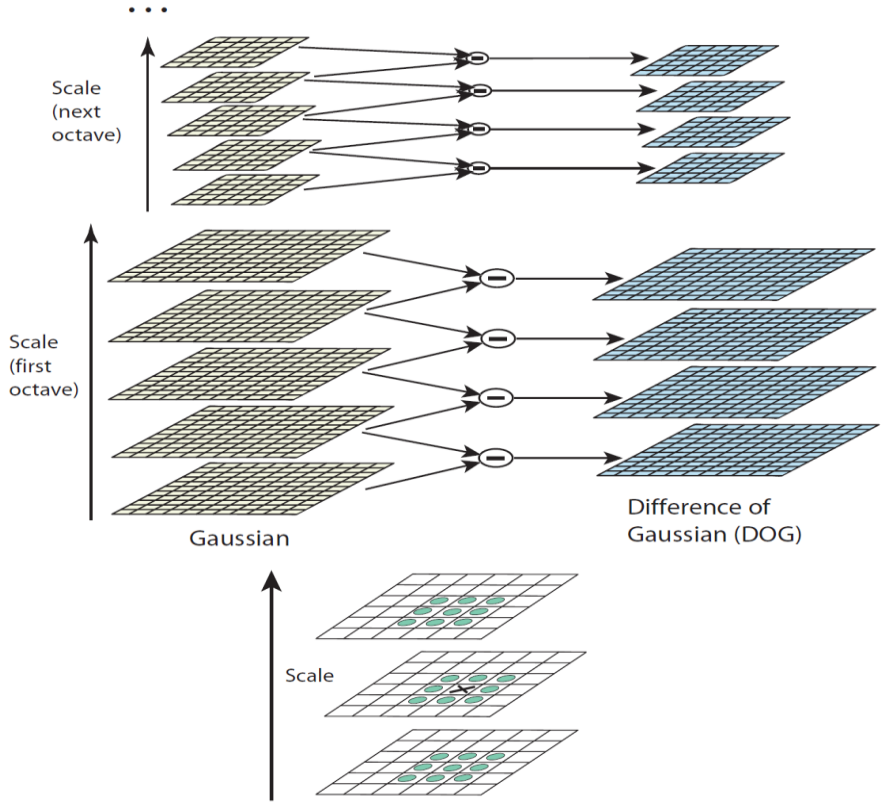
SIFT Algorithm

- Scale Invariant Feature Transform (SIFT)
  - By David Lowe [Lowe2004], ~22.000 citations
  - Extracts a set of invariant features from an image (x/y, scale, rotation, descriptor)
    - Invariant to scaling & rotation, partially invariant to illumination changes
  - One of the most widely used feature descriptors (due to its robustness)
  - Application examples
    - Object recognition
    - Mapping and navigation
    - Image stitching

Images courtesy of [Lowe1999]

SIFT Algorithm
3D DoG Scale-space extrema detection

- Calculation of DoG scale-space
  - For each octave (~ pyramid level)
    - Build a set of gaussian-blurred images with progressively increasing *sigma*
    - Calculate the DoG (difference-of-gaussian) response as difference of two consecutive images in octave
  - DoG operator is approximation of LoG (Laplacian-of-Gaussian) operator
- Detection of local extrema
  - Compare voxel with 26 neighbors in (x,y,s) space
  - Add local maxima/minima to candidate list



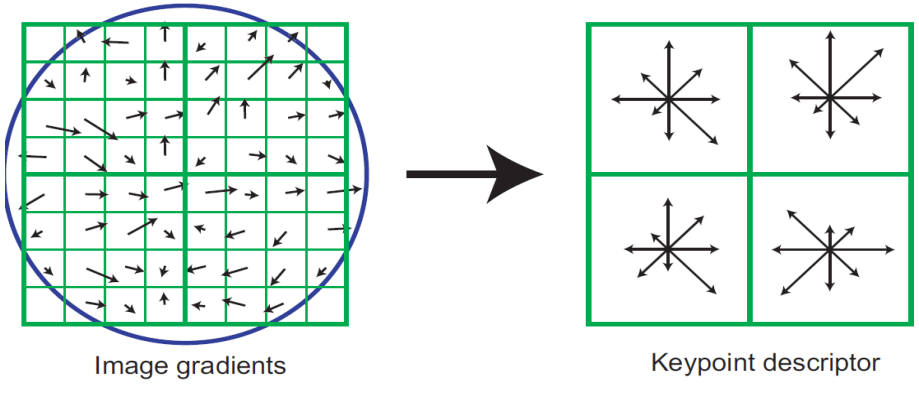
Images courtesy of [Lowe2004]

SIFT Algorithm
Keypoint candidate refinement & filtering

- Accurate localization (for all keypoint candidates)
  - Sub-voxel precise estimation of (x,y,s) coordinate by interpolating a quadratic surface into nearby data points
  - Multiple SVDs (3x3 matrix) have to be calculated for a candidate
- Eliminate edge responses
  - Eliminate keypoint candidates lying on image/object edges
    - As these are unstable e.g. against noise
  - Calculate principal curvatures of 2x2 Hessian (dxx,dxy,dyy) at (**x**,***y***,s)
- Orientation assignment
  - By determining peaks in the histogram of the image gradient orientations

SIFT Algorithm
Keypoint descriptor calculation

- Calculate descriptor of size 128 from image gradient magnitudes and orientations in neighborhood of keypoint
  - 4x4 array of histograms with 8 orientation bins in each
- Several (important) refinements for robustness
  - Gaussian weighting function applied
  - Trilinear interpolation of gradient sample within spatial and orientation bins
  - Illumination normalization of descriptor



Images courtesy of [Lowe2004]

SIFT Algorithm GPU implementation (1)

- Gaussian blurring
  - Standard approach (separable convolution , shared memory usage)
- Detection of local extrema
  - One kernel does all layers of octave + generation of candidate list
    - Move from layer to layer, replaced oldest with new one, cache in shared memory
  - Generation of keypoint candidate list (vector)
    - Done in 3 steps in order to minimize number of simultanous atomic accesses
      - Each thread-block calculates how much space, within the global candidate list, it needs to store its own extrema candidates
      - Each thread-block calculates its own offset in the global candidate list
      - Write all candidates simultanously into the global candidate list

SIFT Algorithm GPU implementation (2)

- Accurate localization (for all keypoint candidates)
  - SVD (of 3x3 matrix) computation uses ideas from [Adams2011]
    - Technical report tries to minimizes branches and uses only elementary floating point operations and inverse square-roots, target is CPU vector unit
    - Jacobi eigenvalue algorithm, applies Givens rotations using quaternions
- Orientation assignment
  - Efficient calculation of the orientation histogram is quite involved
    - Different window sizes for each keypoint, keypoints are scattered throughout (x,y,s) space
    - Solution: **Use one thread block for one key point**
      - Ensures that global memory accesses are coalesced (very important)
      - Also helps to equal out the per-thread work distribution
    - Histogram is kept in shared memory (for Fermi architecture)

SIFT Algorithm GPU implementation (3)

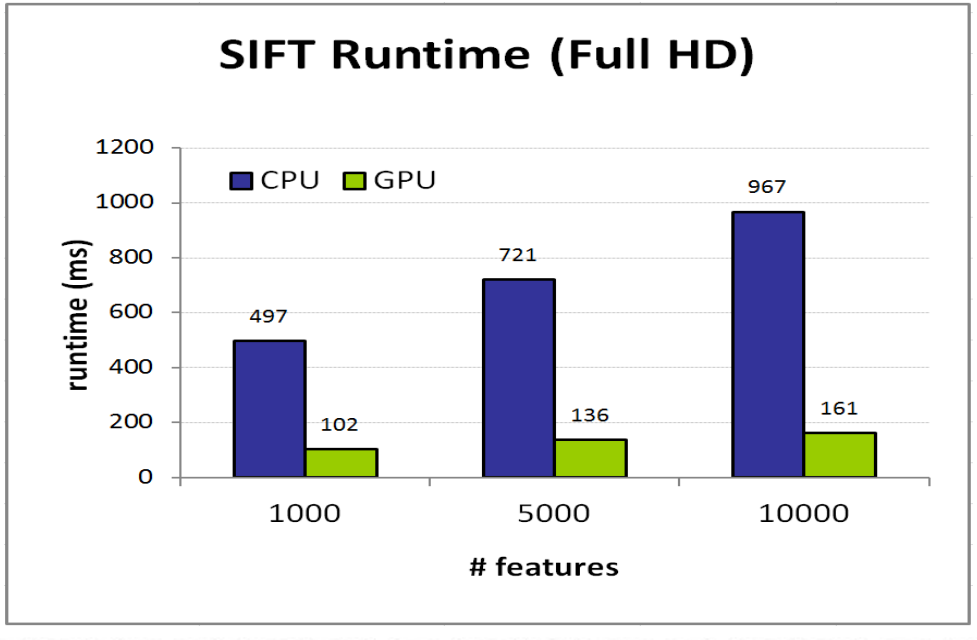
- Histogram computation for keypoint descriptor calculation
  - Similar issues (with regard to global memory accesses) as in calculation of orientation histogram
    - But more arithmetic operations due to larger windows and due to trilinear interpolation
  - Same strategies are used to overcome these issues
    - One thread block per keypoint, histogram is kept in shared memory, …
- Illumination normalization of descriptor
  - L2 normalization, classical reduction

SIFT Algorithm
Runtime & quality comparison (1)

- CPU implementation from HessSIFT library
  - See [HessSIFT2010]
  - Uses only one CPU core, runtime has therefore been divided by factor 4 in order to simulate result for Quad-Core CPU
- Quality tests
  - Nearly identical results when initial image resampling disabled, minor differences when initial bicubic resampling enabled (OpenCV 2.4 does it differently than we)
- Test setup for runtime tests
  - CPU: Xeon 3.0 Ghz Quad-Core, GPU: NVIDIA Geforce GTX 480
  - Initial image resampling (to double size) enabled in CPU/GPU SIFT
  - Total runtime is measured (detector + descriptor calculation), in milliseconds

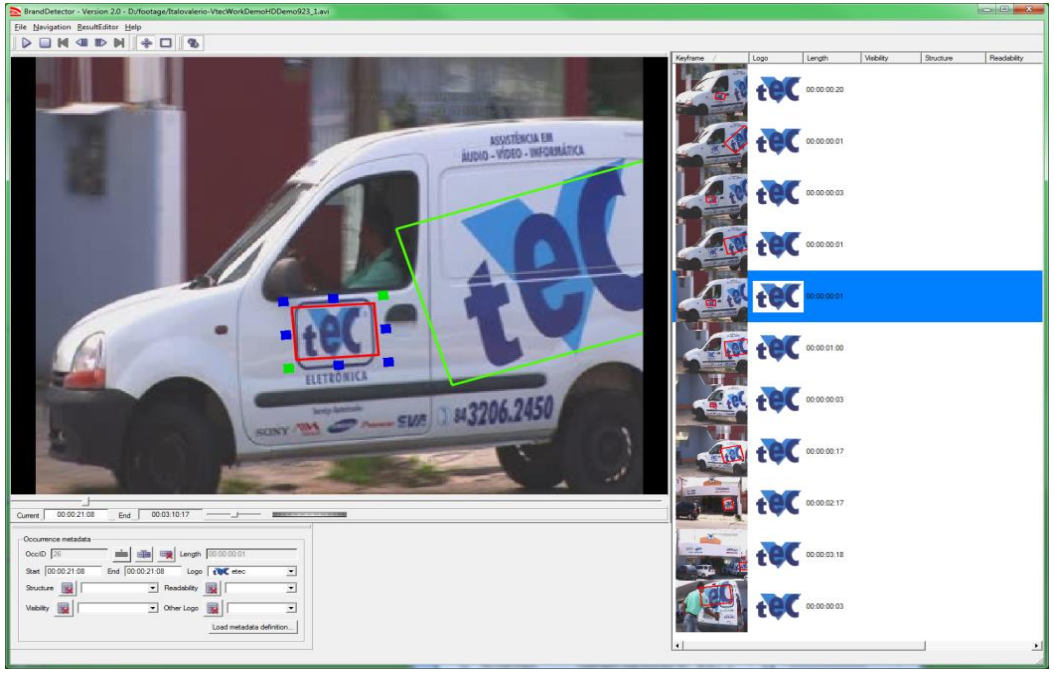
SIFT Algorithm
Runtime & quality comparison (2)

- Overall speedup (GPU vs. Quad-Core CPU)
  - Factor 3 – 5 for SD resolution
  - Factor 5 – 6 for Full HD resolution



Brand monitoring application

- For automatic monitoring of detection and recognition of company logos in TV-broadcast content
  - Provides information about brand visibility in TV
  - Automatic detection of logo occurrences (size, duration)
  - Benefits greatly from GPU-acceleration of SIFT descriptor extraction and matching



BrandDetector application

Near-duplicate video detection

- Framework for clustering of video sections with a similar visual appearance
  - Must work for large amounts of video in audiovisual archives
- Scalable between different degrees of visual similarity
  - Duplicates (e.g., repeated broadcast of the same item, possibly with content transformations applied)
  - Near-duplicates (e.g., similar shot from a very camera position)
    - E.g. multiple takes from the same scene
    - Common situation in unedited raw material (rushes)
  - Partial matches such as picture-in-picture or shots at the same location

Near-duplicate video detection

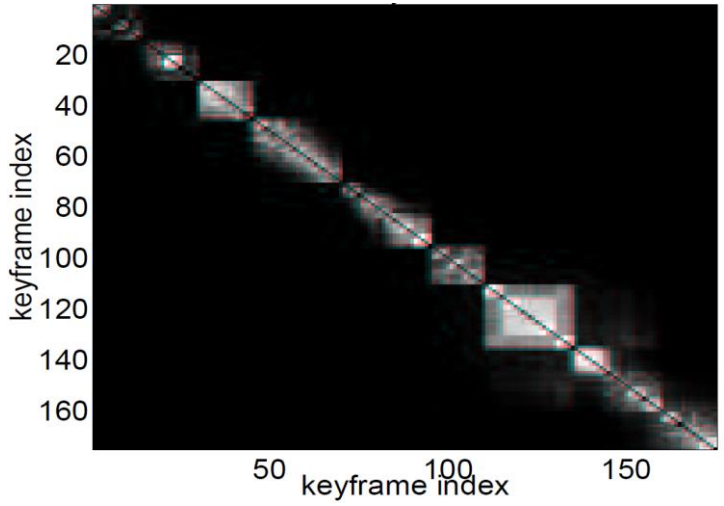
- Examples

Example from MediaEval search & hyperlinking task

Key frames for two takes (upper row, lower row) of a scene. Images courtesy of BBC, have been taken from BBC 2007 rushes video data set (see also Acknowledgments slide).

Near-duplicate video detection

- Workflow of framework for near-duplicate video detection
  - Calculate of features for key frames of the video
    - SIFT (with additional verification step using homographies)
    - MPEG-7 color descriptors
    - …
  - Calculation of affinity matrix
    - Each entry is the similarity score between two individual key frames
  - Search for diagonally oriented regions in the affinity matrix with high similarity values
    - Indicate a ‘match’ between two video sections



Example for an affinity matrix.
Images courtesy of [Odobez2003]

References

- [Adams 2011] A. Adams, A. Selle, R. Tamstorf, J. Teran, E. Sifakis, “Computing the Singular Value Decomposition of 3x3 matrices with minimal branching and elementary floating point operations”, Technical Report #1690, University of Wisconsin-Madison, May 2011
- [HessSIFT2010] HessSIFT library, source code at https://github.com/robwhess/opensift
- [Lowe1999] D. Lowe, Object Recognition from Local Scale-Invariant Features, International Conference on Computer Vision, 1999
- [Lowe2004] D. Lowe, “Distinctive Image Features from Scale-Invariant Keypoints”, International Journal of Computer Vision, 2004
- [Odobez2003] J. Odobez, D. Gatica-Perez, Mael Guillemot, “Video Shot Clustering Using Spectral Methods”, International Workshop on Content-Based Multimedia Indexing, 2003

Acknowledgments

- Silesian University of Technology, Institute of Informatics, Poland
  - Jakub Rosner is a PhD student in project UDA POKL-.04.-01.01-00-106/09 supported by the European Union from the European Social Fund.
- JOANNEUM RESEARCH, DIGITAL – Institute for Information and Communication Technologies, Austria
  - Hermann Fürntratt, Werner Bailer, Harald Stiegler
- BBC 2007 rushes video is copyrighted. The BBC 2007 rushes video used in this work is provided for research purposes by the BBC through the TREC Information Retrieval Research Collection.
- This work was supported by the European Commission under the grant agreement no. FP7-610370, „ICoSOLE“ and under grant agreement no. FP7-287532, “TOSCA-MP – Task-oriented search and content annotation for media production”.
  - http://www.icosole.eu
  - http://www.tosca-mp.eu

Hannes Fassold

hannes.fassold@joanneum.at

JOANNEUM RESEARCH
Forschungsgesellschaft mbH

Institute for Information and
Communication Technologies

www.joanneum.at/digital