Implementation of MPEG-7 visual descriptors in Java

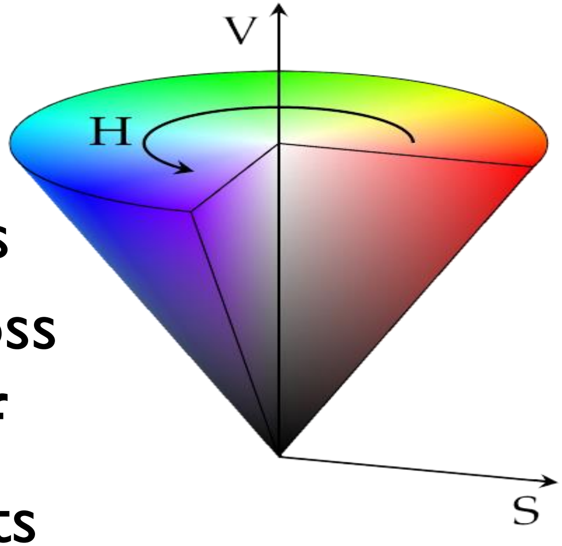
Monika Kostolná

Objective

- Reimplementation of chosen MPEG-7 descriptors in Java
  - Scalable Color Descriptor
  - Edge Histogram Descriptor
- Application of MESSIF framework (DISA)
- Using potential of java libraries
- Comparison with C++ Extractors

Scalable Color Descriptor

- Represents the color element in the image
  - HSV scheme
- Creation of the uniform histogram
- Haar transformation
  - Sums ~ low pass filters
  - Differences ~ high pass filters
  - Compression with no data loss
- Scalability in the context of
  - Number of output coefficients
  - Number of bit planes discarded



Example

- Inputs or outputs can be different
  - The uniform histogram creation
  - The length and the bit depth of the output are defined by coefficients

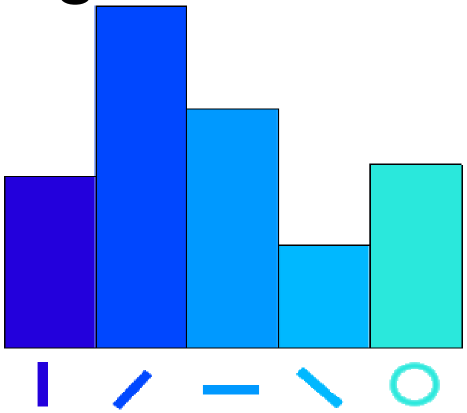
-13,3,6,15,1,1,2,3

-20,6,3,10,1,1,2,3

- *Example:*
  - *NumberOfCoefficients = 8*
  - *NumberOfBitPlanesDiscarded = 3*

Edge Histogram Descriptor

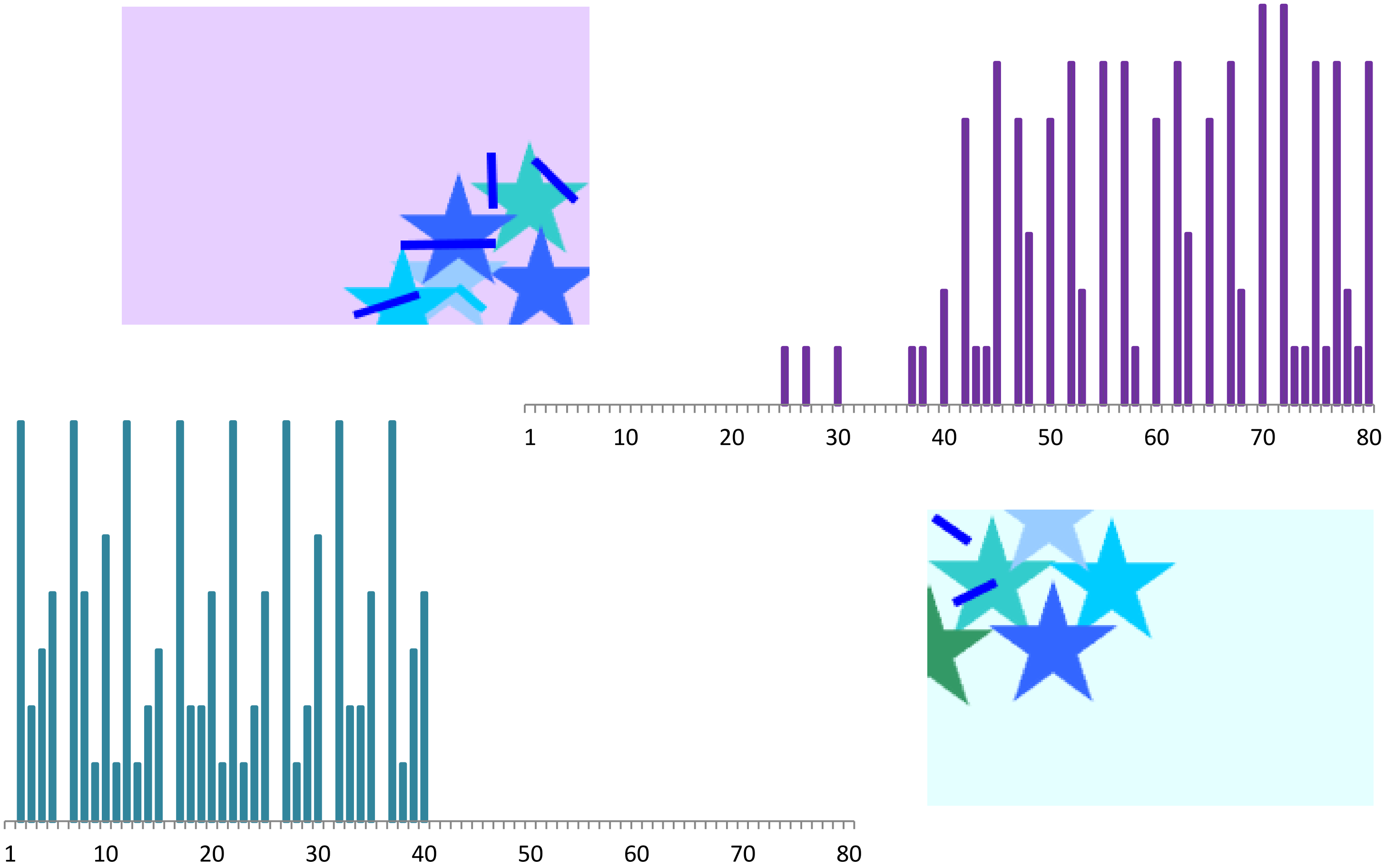
- Represents the texture and the shape elements in image
- Input divided into 4x4 sub-images
  - For each sub-image is calculated an histogram with 5 types of edges
    - Vertical
    - 45-degree diagonal
    - Horizontal
    - 135-degree diagonal
    - Non-directional



- Final histogram of 80 bins (4x4x5)

Example

- 2 images with different locations of edges



Advantages of Java implementation

- Usage of Metric Similarity Search Implementation Framework (MESSIF)
  - Simplification of the work in metric space
  - Direct connection to application using MESSIF
  - Used for
    - Comparing C++ and Java implementation on andromeda
    - Extraction output
    - Testing

Advantages of Java implementation

- ImageJ API
  - Open source program
  - ImagePlus, ImageProcessor
    - Contains pixel data of 2D picture
  - ImageConverter
    - convertToGray8
- Maven
  - Tool for
    - Dependency management
    - Project building
  - Dynamically downloads required java libraries

Comparison with C++ extractor

- Portability
  - C++
    - Platform dependent build required
  - Java
    - Platform independent (portable)
      - Creation of native code executable on any machine
- Code readability
  - Original C++ source code
    - Very long and difficult to maintain
  - Created Java source code
    - Simple and documented methods
    - Conventions following

Comparison with C++ extractor

- ScalableColor
  - 50 small images
    - C++: 613 ms
    - Java: 775 ms
  - 50 big images
    - C++: 3420 ms
    - Java: 5506 ms
- EdgeHistogram
  - 50 small images
    - C++: 203 ms
    - Java: 280 ms
  - 50 big images
    - C++: 822 ms
    - Java: 1248 ms
- Future optimization possibilities

Thank you for your attention.