

LUX color transform for mosaic image rendering

Franck Luthon, Prof. ⁽¹⁾

Brice Beaumesnil, Dr. ⁽²⁾

Nicolas Dubois, M.Sc. ⁽²⁾

I.U.T Bayonne-Pays Basque
Dept. of Industrial Engineering ⁽¹⁾
Computer Science Dept. ⁽²⁾

University of Pau and Adour,
Anglet, France

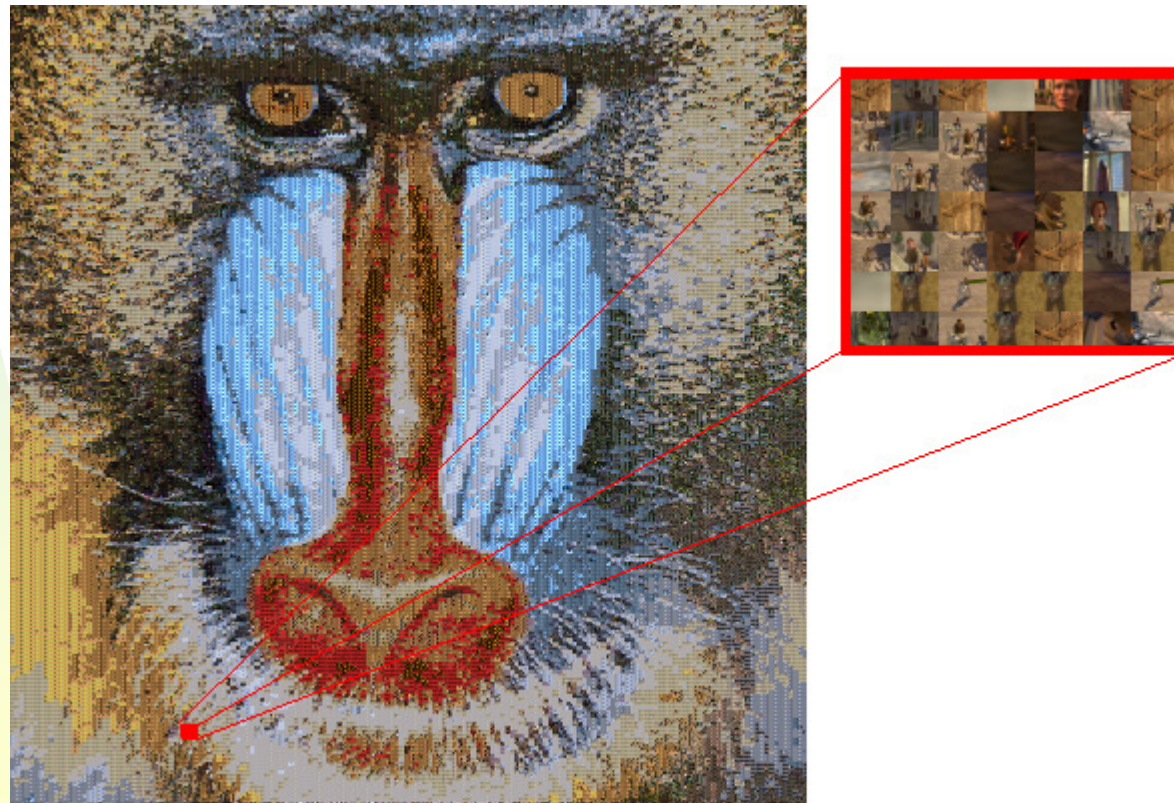


Outline of the talk

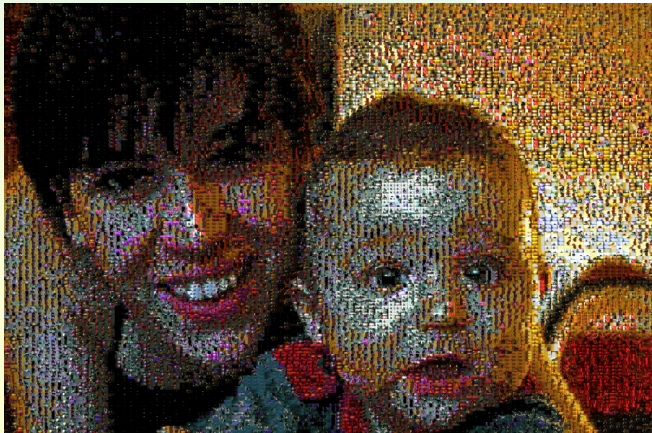
- Principle of mosaic image
- Color matching
 - ◆ improved color rendering
- Computationally efficient algorithm
 - ◆ reduced computation cost
- LUX nonlinear (log-like) color transform
 - ◆ color adaptation (bio-like) behavior
- Experimental results (PSNR comparison)
- Discussion & future work

Introduction

- Photomosaic made of little frames from a video
 - ◆ 2 inputs: one target image + one video
 - ◆ output: one mosaic image
- zoom:



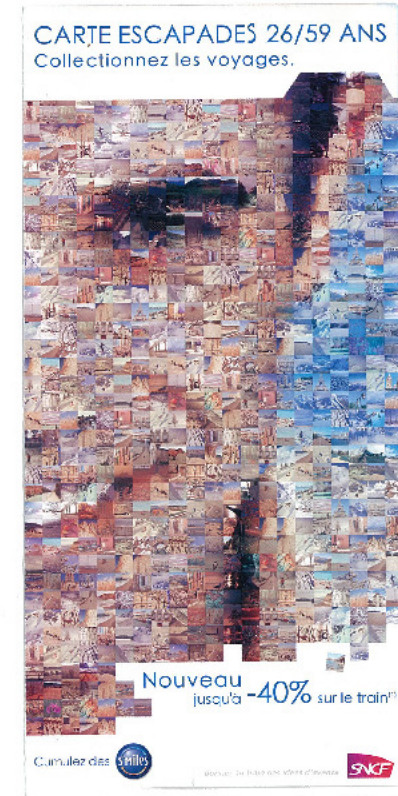
Applications



AQTR 2010 - Cluj

- Commercial
 - ◆ Advertising
 - ◆ Cover pages
 - ◆ Posters...

- Artistic
 - ◆ Postcards
 - ◆ Personal photo album
 - ◆ Graphical art
 - ◆ Computer graphics...

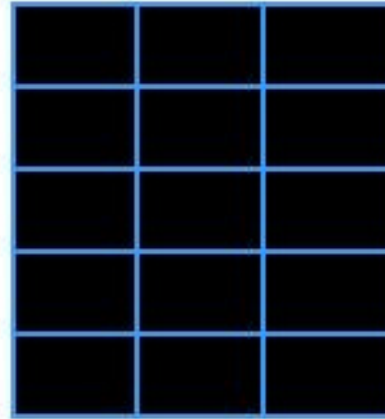
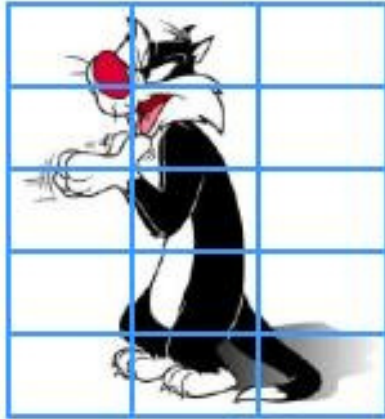


LUX color transform, Luthon et al.

Key issues

- Color matching step
 - ◆ to improve color contrast
- => choice of a good color space
- Aesthetical effect
- Computation cost
- => simple & efficient algorithm

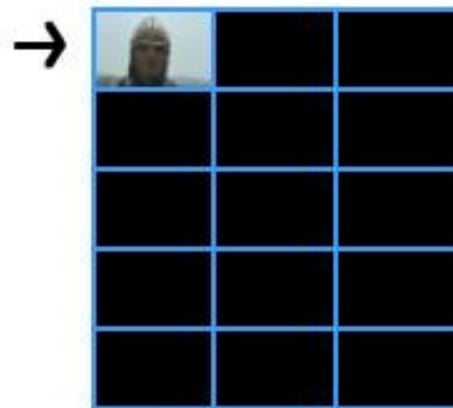
Principle of mosaic



200	200	200
200	200	200
200	200	200
200	200	200
200	200	200

Distance(, ) = 156

■ => Improvement : difference = 44



→

156	200	200
200	200	200
200	200	200
200	200	200
200	200	200

Optimization algorithm

- For faster rendering : reduce iteration number
- Discard « bad frames »
 - ◆ L.U.T. to store color distances
 - ◆ Quality criterion:
 - ☞ too big distances (compared to current ones in LUT)
 - ◆ Shot detection (in video at 25 img/sec)
 - ☞ Reject half sec (13 frames) following a bad frame
- Place « good frames »
 - ◆ Unicity constraint
 - ◆ Quality constraint (color likeness) => best fit ?
 - ◆ First found (then exit comparison loop)
 - ◆ Best (lowest) distance (local criterion)
 - ◆ Best (biggest) difference (global improvement)
 - ◆ Stack of candidates (memorize alternative places):
 - ☞ Recursive: reevaluate temporarily removed frames
 - ☞ Optimal : best place (through replacement)

Color matching

- Distance

$$d(A, B) = \sum_{i=1}^3 \sum_{n=1}^N |A_i(n) - B_i(n)|$$

- Linear color spaces like YUV, YCrCb...

$$\begin{bmatrix} Y \\ Cr \\ Cb \end{bmatrix} = T \cdot \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

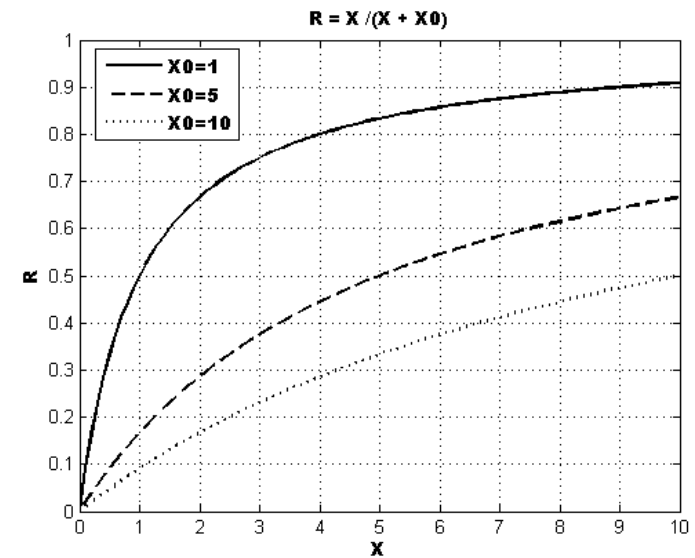
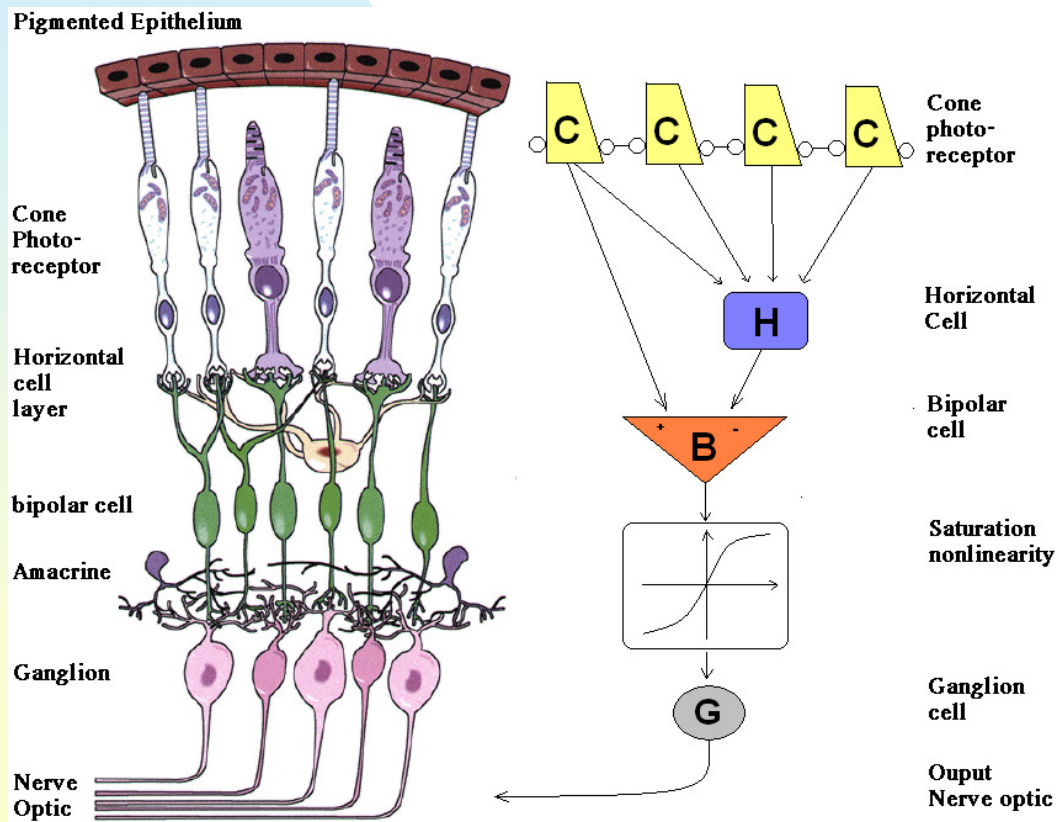
- Luma-Chroma Matrix T

$$T = \begin{bmatrix} 0.3 & 0.6 & 0.1 \\ 0.7 & -0.6 & -0.1 \\ -0.3 & -0.6 & 0.9 \end{bmatrix} \quad \sum_{j=1}^3 t_{1j} = 1 \quad \sum_{j=1}^3 t_{2j} = 0 \quad \sum_{j=1}^3 t_{3j} = 0$$

- Nonlinear color spaces: HLS, CIE...

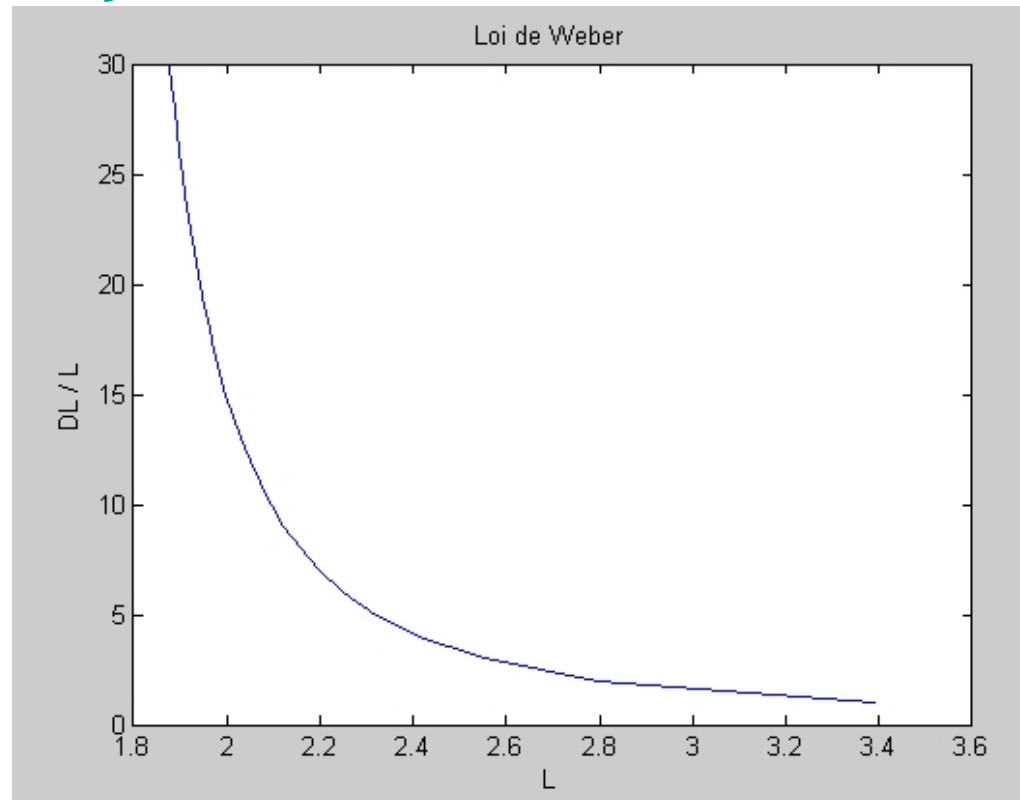
LUX color transform background: (1) biological analogy

- Biology : cone transduction in retina is log-like



Human Visual System (HVS) cf. Weber law

- Local contrast of luma: $\Delta L/L$
- « just noticeable difference »



LUX background: (2) mathematical LIP model

- LIP model for greyscale (where $x_0 = \max(x)$; $M=256$)

$$\Phi : x \rightarrow y = M \ln\left(\frac{x_0}{x}\right)$$

$$\Phi^{-1} : y \rightarrow x = x_0 \exp\left(-\frac{y}{M}\right)$$

- LUX model for luma+chroma: RGB \rightarrow log() \rightarrow T \rightarrow exp()

$$(R, G, B) \xrightarrow{\text{norma}} (\bar{R}, \bar{G}, \bar{B}) \xrightarrow{\Phi^{-1} \circ T \circ \Phi} (l, u, x) \xrightarrow{\text{denorma}} (L, U, X)$$

- Acronym: Logarithmic hUe eXtension

LUX implementation details : 3 main steps

- **Normalization** of RGB components
 - ◆ To enable computation of $\log(\cdot)$
- **Application of composition:** $\Psi = \Phi^{-1} \circ T \circ \Phi$
- Test on chroma dynamic range
 - ◆ To account for negative chroma values
- Proper **denormalization**
 - ◆ To get final color components in $[0;255]$
 - ◆ Inverse transform exists: LUX \rightarrow RGB

LUX direct computation

- Simplified formulae:
 - approximation
 - with $\max(.) = \text{cst} = M$
-
- where $M=256$

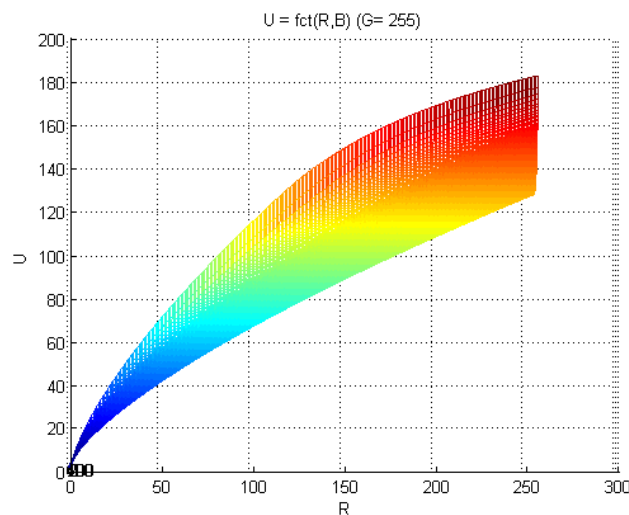
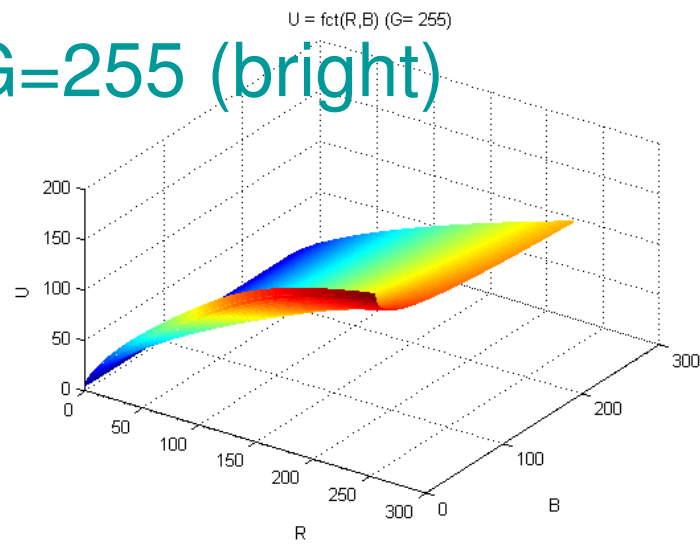
$$L = (R + 1)^{0.3} (G + 1)^{0.6} (B + 1)^{0.1} - 1$$

$$U = \begin{cases} \frac{M}{2} \left(\frac{R+1}{L+1} \right) & \text{if } R < L \\ M - \frac{M}{2} \left(\frac{L+1}{R+1} \right) & \text{if } R \geq L \end{cases}$$

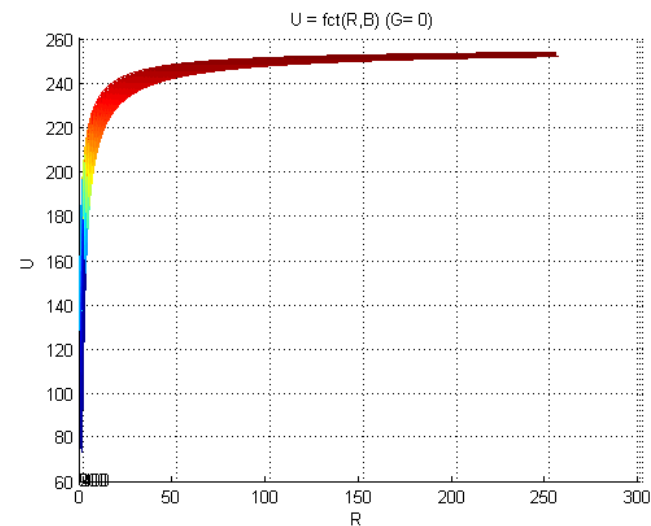
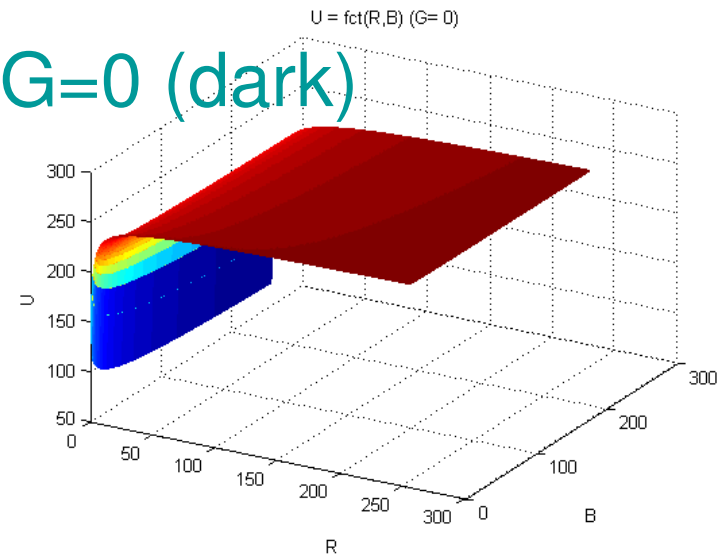
$$X = \begin{cases} \frac{M}{2} \left(\frac{B+1}{L+1} \right) & \text{if } B < L \\ M - \frac{M}{2} \left(\frac{L+1}{B+1} \right) & \text{if } B \geq L \end{cases}$$

Logarithmic curves $U(R,G,B)$

- $G=255$ (bright)



- $G=0$ (dark)



Prototype

- Linux / Windows: GTK+OpenCV
- Computation cost:
 - ◆ 8ms / tile of size 20x15
 - ◆ <1min / img of 60x80 tiles (i.e. size of 1200x1200 pixels)
- Various color spaces (for computing distances):
 - ◆ Linear: RGB, YCrCb, XYZ
 - ◆ Nonlinear: HLS, HSV, CIELab, CIEluv, LUX
- PSNR: computed in RGB color space

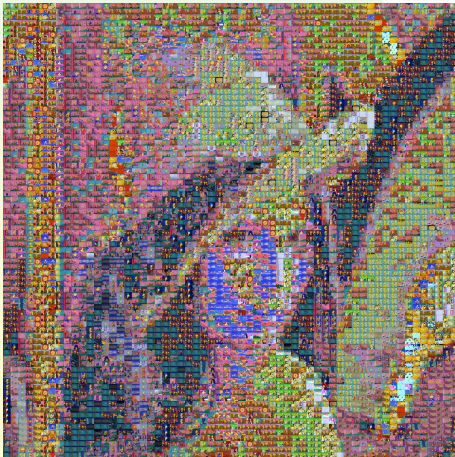
$$PSNR(A, B) = \frac{1}{3} \sum_{i=1}^3 \frac{255^2}{\frac{1}{N} \sum_{n=1}^N [A_i(n) - B_i(n)]^2}$$

Example 1: Lena

- Original

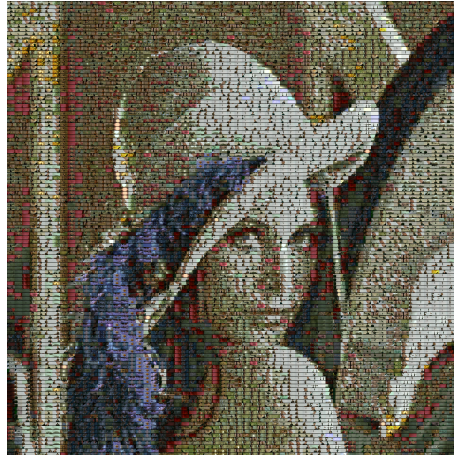


- HSL 22.4dB

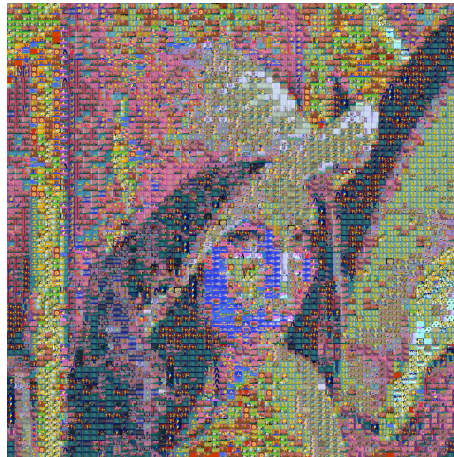


AQTR 2010 - Cluj

- YCrCb 22.2dB

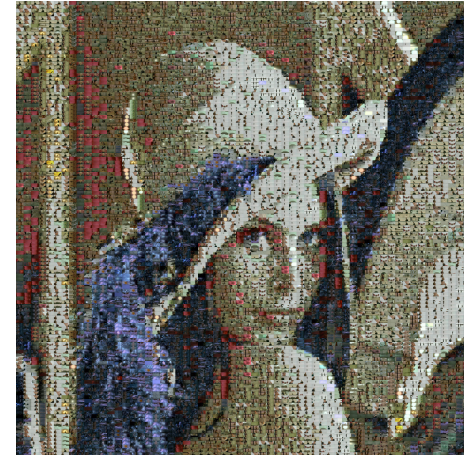


- HSV 23dB

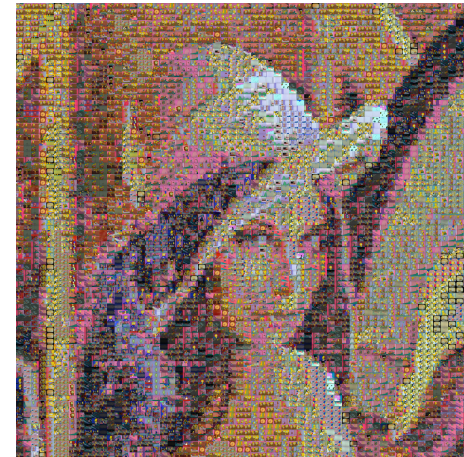


LUX color transform, Luthon et al.

- LUX 25.3dB

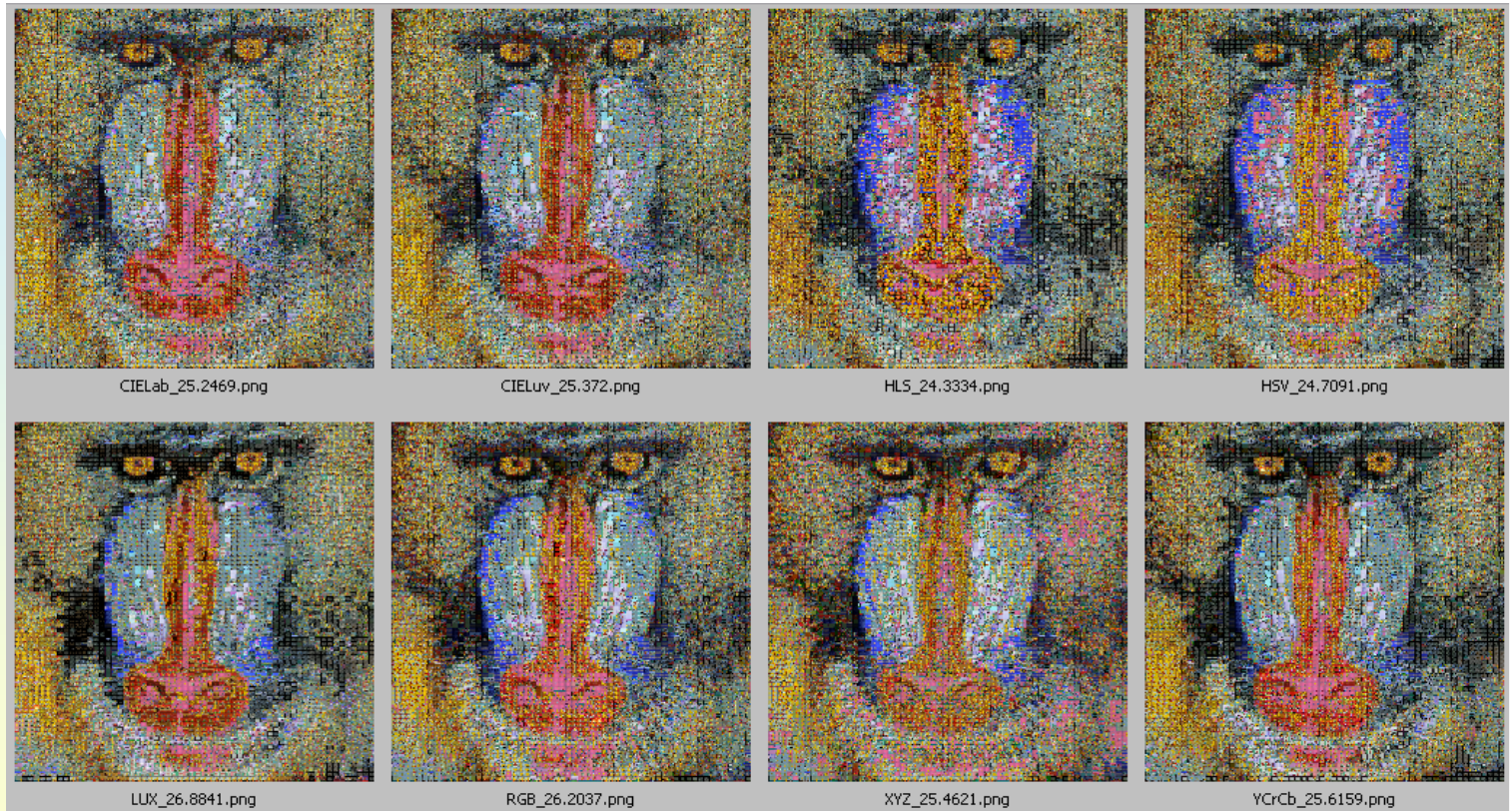
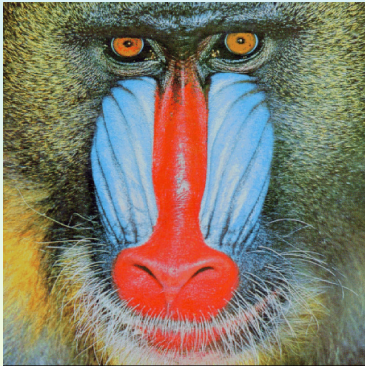


- LUX 26.4dB



Example 2: Mandril

- Best PSNR: LUX=26.9dB



- Good contrast

Example 3: Shrek



- Higher PSNR when target image is one of the frames of the video
- Here: LUX 31 dB



Evaluation results

- PSNR : LUX > other color spaces
 - ◆ +1dB to +3dB
- Better qualitative color rendering
 - ◆ HLS, HSV: poor color match (bluish)
 - ◆ CIE Lab, CIE Luv: low contrast
 - ◆ XYZ: pale rendering
 - ◆ RGB: variable performance (smudge)
 - ◆ YCrCb: good but PSNR < LUX
 - ◆ LUX: better contrast

Discussion & Future work

- Further theoretical investigation on LUX transform
- Qualitative comparison
 - ◆ Mean opinion score measure
- Advanced mosaicing
 - ◆ Variable tile size (larger or smaller depending on content)
 - ◆ Variable shape
 - ◆ Shot detection in video
 - ◆ Other color distance (e.g. joint color histograms)
- Other applications
- Video-mosaics: real-time implementation with GPU