IIIF Imageserver

*a new way to present images on the web*

Prof. Dr. Lukas Rosenthaler
Digital Humanities Lab
University of Basel
Referencing images on external sites is difficult!

- each provider has a different implementation
- HTTP-Links (=url’s) provide just one version (format, resolution, region)
  - problem client-side and/or server-side
  - either: only very few versions available
  - or: complex handling on server which has to keep many versions
- usually no systematic way to construct URL
IIIF-Solution

- Standard URL which allows to define
  - image resolution
  - image region
  - image format

{scheme}://{server}{/prefix}/{identifier}/{region}/{size}/{rotation}/{quality}.{format}
{scheme}://{server}/{prefix}/{identifier}/{region}/{size}/{rotation}/{quality}.{format}

- **scheme**: http or https
- **server**: DNS name
- **prefix**: arbitrary name
- **identifier**: unique identifier of image
- **region**: “full” or region in pixel coordinates
- **size**: “full” or pixels
- **rotation**: angle (degree) and mirroring (“!”)
- **quality**: “default”, “color”
- **format**: “tif”, “jpg”, “png”, “jp2”

http://www.salsah.org/limc/z412_23a/100,400,800,300/pct:50/90/default.jpg
IIIF-Server

• parse URL
• prepare image “on the fly”
  ‣ cropping
  ‣ scaling
  ‣ rotating
• send image using HTTP protocol
SIPI (simple Image Presentation Interface):

- an expanded implementation of a IIIF-server (high efficient stand-alone server written in C++)
- Implements the full specification of IIIF
- simple configuration
- additional features:
  - fully scriptable with Lua (pre-flight, in-flight, post-flight) access control, database connection
  - embedded simple web server (elua = embedded lua)
  - restricted views “on the fly”: watermarks, resolution restrictions etc.
SIPI (Simple Image Presentation Interface):

- supports uploads, import and format conversions (controlled by Lua scripts)
- master images stored in highest resolution (size, bit-depth) as JPEG2000
- supports CORS (Cross-Origin Resource Sharing)
- uses kakadu-JPEG2000 library (license required!)
- preserves all metadata (TIFF-tags, EXIF, IPTC, XMP and ICC color profiles)
- supports tiled viewers (e.g. openseadragon) for high-resolution image viewing on the web
- open source (GNU Affero General Public License), on github (https://github.com/dhlab-basel/Sipi)
What is JPEG2000

• **NOT** a enhancement of JPEG
  it’s a completely new compression method

• based on **wavelets**

• a **standard** with a lot of options and variants (*which makes it difficult to master from time to time…*)

• base for the IIIF compatible Imageservers
Wavelets: e.g. Gabor-Filters

With $\phi$ denoting the rotation angle

\[
u(x, y) = x \cdot \cos(\phi) + y \cdot \sin(\phi)\]

\[
v(x, y) = -x \cdot \sin(\phi) + y \cdot \cos(\phi)\]

We get the filter pair with a given $\sigma_x$, $\sigma_y$ and frequency $\nu$

\[
G_e(x, y, \phi) = \cos(\nu) \cdot e^{-\frac{1}{2} \left( \frac{\nu}{\sigma_v} \right)^2} \cdot e^{-\frac{1}{2} \left( \frac{u}{\sigma_u} \right)^2}
\]

\[
G_o(x, y, \phi) = \sin(\nu) \cdot e^{-\frac{1}{2} \left( \frac{\nu}{\sigma_v} \right)^2} \cdot e^{-\frac{1}{2} \left( \frac{u}{\sigma_u} \right)^2}
\]
„local energy“

$$A(x, y, \phi) = \sqrt{G_e(x, y, \phi)^2 + G_o(x, y, \phi)^2}$$

- Response to an optimal line or edge

$$l(x, y) = e^{-\frac{1}{2} \left( \frac{x}{\sigma_v} \right)^2}$$
Brain equivalent

• even and odd Gabor => "simple cells"
  • respond to lines or edges
  • orientation selective

• "local energy“ => "complex cells"
  • respond the same to lines and edges
  • orientation selective
Wavelets...?

• are "natural" because the vision and segmentation process of the brain is based on wavelets (visual cortex V1 & V2)


http://iiif.io