Ajapaik – crowdsourcing platform for geotagging and rephotographing historic pictures

Ajapaik is an open source crowdsourcing platform (built on python django) for enriching historic view images with data (mainly locational data). Currently the platform is very low tech in the sense that it does not use image or text analysis and all the work is done by users manually in a distributed manner: somebody curates the content by making searches in the public collections, selecting images and adding these to (location specific) albums, other users can geotag individual images to the exact spot on the map where photographer was standing when taking the picture, next group of users can make rephotographs of historic pictures creating then and now picture pairs etc.

There are more than 110 000 pictures (mainly photographs, but also paintings and graphic art) on the platform and almost 61 000 of these have been geotagged by users (see graph). There are more than 5300 rephotographs. Our data set has drawn attention from abroad. We are currently preparing a Horizon2020 application with Joanneum Research and 3 other partners. Recently we were contacted from Ghent University where a research group is also working on algorithms for detecting places and attributing dating to historic images.

Ajapaik is officially a present for EV100. See also slides from python codeclub (autumn 2016).

Project #1: Applying image analysis

Duplicate detection

Photography was a reproducible media already in the analogue era therefore different exemplars of same images exist in photographic collections (for instance one museum might hold a glass plate negative, another has a printed postcard of the same image) that are currently not linked to each other. When considering the geotagging of visual content duplicates of same image should be grouped together. Currently Ajapaik uses no computer vision or image analysis for grouping pictures together, implementing this would significantly improve the organisation of images.

Object detection, automated tagging

Next level would be detecting same objects (buildings) on different images.

Image analysis features have to be implemented in both backend and frontend (user interface) where users will validate the true positives and remove wrong results.

Project #2: Text analysis, albums' system and GIS

As the graph demonstrates the overall number of images is growing faster than the number of images geotagged by users. So we need to put more effort into detecting the approximate location from the image metadata (description in the museum collection). The descriptions vary a lot, sometimes there are even exact addresses in the description (although it's rarely the case), usually there is some kind of approximate description of location (village or town name). There are also cases where several place names might be mentioned in the description, for instance "Põlva laulukoor üldlaulupeo rongkäigus Tallinnas" ("choir from Põlva in the parade of the singing festival in Tallinn").

We need to detect possible place names and show this kind of approximate location in the user interface (on the map), engage users in confirming the approximations and also get them submit the exact location.

Currently the GIS in Ajapaik is very simplistic: we have albums with location (attributed from Google Maps Places API) and images with exact geotags. The album location only sets the geotagging interface in the mentioned location, but images of album-level location attribution do not show up on the map. Also there is just two levels in the albums: an album can be a sub-album of another album, but actually the location-based albums should have much more levels and be self-populating: an image should show up in the city district album, in the city album and in the county album etc. Currently when an image is in a typological album (for instance 'ortodox churches' or 'schoolhouses') and gets geotagged in a specific village or town it won't show up in the album of this locality.

Project #3: Social validation algorithms

Having an interface for users to submit geotags for images is only the first and easy step. Much more difficult is dealing with conflicting suggestions for a same image. If one user suggests on location and another suggest a different location then who do we trust?
In the very first version of Ajapaik there were only two views: a map view of geotagged images where user could only see the images and a geotagger interface where user got images from a specific town album one by one and had to submit a location (geotag) without seeing what (if any) location had other users submitted for the image.
In this mode we calculated the trustworthiness of the user depending on how much did his/her geotags coincide with previous users geotags. Also the amount of contribution helped to raise the trustworthiness and thereby also the weight of users submission.
Later on we've changed the interface a lot: the main view for albums is now the gallery view, there's a minimap next to the images with the current location. In every view user can submit a new geotag etc.
The idea is that location of an image is always dynamic and all the submitted locations are saved and clustered. If a wrong location is submitted at first, then a new suggestion to another place might not change the location yet, but when a third submission confirms the new location

the location should change. Still we've seen that if a very trustworthy user has submitted a wrong location it's hard to change this by other users.

There are two main goals in solving this problem:
1) adding more social features to the platform that would enable more interaction and discussion between users and displaying submitted information in a better way. If a user has submitted a location for an image and another user suggests a different location the first user should get a notification to review his or her opinion: will s/he want to defend his opinion or agree with the correction. The fact that an image has conflicting geotags should also be displayed to other user in order to engage them into confirmind one or other location etc
2) the algorithm giving weight to submissions should be more complex and simpler at the same time. The trustworthiness of the users should also be recalculated not only based on their own submissions but also retroactively when a suggestion that at the time of its submission had more weight proves wrong later. Instead of continuous value of trustworthiness maybe discrete classes would make more sense (for instance 1–4: a 4-weight geotag could be changed already with 2+3, or 3x1 + 2 etc).

Project #4: Rephotography apps for Ajapaik

We are about to release Ajapaik's new version of rephotography application for Android. The app's main goal is making it easy to take rephotographs (now-pictures next to historic then-pictures). The app displays images nearest to user, the user choose an image and it is displayed semi-transparently on top the camera stream in order to make the juxtaposition of old and contemporary as convenient as possible. The old image can also be zoomed or flipped (sometimes historic images scanned from film can be horizontally flipped).
1) we would also like to have an iOS version of the app in order to announce a wider rephotography campaign next year when Estonia is celebrating its 100th anniversary
2) the app could have much more complexity in both Android and iOS version. The version soon to be published (code on Github) does not have a map view for instance, but only shows a grid of images. There could be the option of also manually correctin the geotag of an images in the app, as being really on the street and looking for the definite vantage point of the old image is always more precise than the estimation made on the map behind a desktop. There could be routing options to make it easier for users to choose images for rephotography as in some places the density of historic images is already too high and can thus be disturbing while an option to get a route for a 3 km walk with 10 images on the road to be rephotographed would be much nicer etc.

IP

All contributions during the project(s) will be published under an open source licence in the Ajapaik Github Repository.