

Jonas Luft¹

Automatic Georeferencing of Historical Maps by Geocoding

Keywords: georeferencing; geocoding; label extraction; place-names; historical maps

Summary: Libraries and researchers face a big challenge: sometimes thousands, millions of maps are being conserved in archives to retard deterioration, but this makes it hard to obtain specific knowledge of their content, sometimes even of their existence. In recent years, the efforts have increased to digitise historical documents and maps to preserve them digitally, make them accessible and allow researchers and the general public a less restricted access to their heritage. But maps are more than a cultural artifact: they are data. Data about the past that can very well be important for science and decision-making today. The problem is, the vast amount alone makes it hard to know what to look for. Maps are usually archived and catalogued with limited meta-information, sometimes obscuring their actual content. This makes it impossible to find specific information without expert knowledge on history and cartography. Extracting the content, i.e. data, of scanned historical maps is a necessity to make the content searchable and to use modern digital tools for automatic processing and analyses. To combine data from different maps and compare them with modern geospatial data, georeferencing is paramount. This is usually done by hand and needs a lot of time and specialised training. We explore if and how usable GCP can be found automatically in historical maps with computer vision and document analysis methods of today. In this work-in-progress report, we use OCR on map labels and geocoding of persisting geographical feature designation for successful first experiments. This shows text recognition and vectorisation to be a promising research direction for large-scale automated georeferencing of historical maps.

Introduction

Until only few years ago almost any motorist used a road atlas to navigate. They usually contained an index of cities, road names or other destinations along with their corresponding location in the atlas and the respective map sheet. This index, also called a *gazetteer* in other map types, enables a quick localisation of named places in the map's respective reference system. Recently, gazetteers in this form have become somewhat less common. Instead, we now have access to a plethora of geocoding web-services which can be queried with search terms to identify and localise almost any location all over the world.

Today, historical maps are catalogued according to their metadata and not their content. To make spatial queries (in the form of "all 19th century maps showing Hamburg") or spatial analyses possible, they have to be georeferenced (Schröder 2013). To georeference a scanned map, matches between a set of so called *ground control points* (GCP) in the image and actual geographical locations have to be found.

¹ CityScienceLab, HafenCity University Hamburg [jonas.luft@hcu-hamburg.de]

After automatically extracting and recognising place names from a map and querying their respective locations in aforementioned geocoding services to obtain geocoordinates, we are able to automatically georeference scanned historical maps. The goal is to thus enable batch processing of large series of maps, such as late modern period land surveys. To make automatic handling feasible with minimal human input, homogeneous graphical quality and design is required. Therefore we also investigate which properties of maps are necessary to allow this approach to be used.

Related work

Automatic matching of similar features in different images is a classic task of computer vision and has been thoroughly researched for spatial data in photogrammetry and applied to historical data (Chen et al 2016). Of special interest is automatic registration of heritage aerial images with modern day data, which are hard to match, since they use different colour spaces (black and white vs. colour images), often have quite different quality (analog vs. digital images, old images scanned from photographic materials), but especially because a significant change in the environment and built structures have to be expected. To overcome these issues Cléry et al. (2014) use line feature matching to register old aerial images with current georeferenced orthoimages, to georeference the historical data.

Interesting research has been done by Chen et al. (2004) into registering orthoimages with vector data, by automatically extracting road intersections from the raster image and matching those to vector data of the same scene for very high precision results.

To generalise this approach to unknown scenes Briggs and Li (2006) employ topological point pattern matching to find road intersections extracted from a raster map in a much larger dataset of vector data.

Wolter et al. (2017) have proven that place names in maps are very descriptive and can be used for identification and localisation of settlements and river tributaries. By using spatial reasoning, they have developed a method to automatically identify a river network in a map. So far there has been limited research into automatic methods for extraction of control points to be used in registration and georeferencing that take the special difficulties of early modern era maps that are hand-drawn or lithography-printed, into consideration. In one prime example (Heitzler et al. 2018), the map boundary and grid lines are automatically detected by template matching and Hough line detection and subsequently matched with coordinate labels to automatically rectify and georeference the Siegfried map.

It remains open, how to deal with maps that don't show grid lines, are too cluttered to extract them successfully, are drawn with unknown projection and datum or are distorted so extremely that given coordinates can't be trusted.

Problem Definition

For most computer vision tasks, high-contrast points in images are used to build robust feature descriptors, which have over the years been the basis of increasingly better performance on detection and classification tasks. Increasingly complex object representations enabled by machine learning accurately recognize objects in cluttered scenes. The special property of maps on the other hand, which distinguishes them from almost every other computer vision task, is that they have already been optimised for human use. In order to improve readability of maps

in spite of a high density of information, mapmakers have developed abstract and generalised coding, almost completely removing latent information from the image texture. The steep gradients on the variety of overlaid lines, symbols and region boundaries make a map a homogeneously high contrast, high clutter image with ambiguous repeating patterns prohibiting the use of state-of-the-art computer vision descriptors for uniquely defining objects.

The biggest challenge for the application computer vision methods on topographic maps is that the possible figure of interest (i.e. foreground) exhibits similar properties to the (back-) ground. In particular, when we are interested in text content, it is very hard to distinguish from the remaining content of the map, such as contour lines, roads, symbolic features and building outlines.

A first step for figure-ground-segmentations can be the use of colour information to separate multiple feature layers (e.g. bodies of water in blue, contour lines in brown, writing in black) of the map. But since maps are usually drawn with only a limited palette of colours this will still result in overlapping features. This is an expected problem especially with old maps produced in lithography. Since creation of multiple printing plates immensely increased the production cost without significantly increasing readability, it is quite common to see maps with only three to four colours.

The resulting ambiguous use of (most commonly) black colour for writing as well as grid lines, special symbols and building outlines can make a complete figure-ground-segmentation impossible and severely impairs the quality of *optical character recognition* (OCR) results.

Machine-reading in maps poses some unique problems as a result of the apparently arbitrary positioning of text labels. While document analysis methods can usually rely on homogeneous line and character orientation and spacing to resolve ambiguities in character recognition, the page segmentation on maps is much harder: we have to deal with multiple fonts in multiple sizes, sometimes widely spaced and overlapping each other; text labels can have unusual orientations and even be curved to follow e.g. the path of a river. All of these make it especially difficult to know which two (or more) labels belong together when a designation has been split over multiple “lines” and placed in seemingly unrelated locations.

Finally, there are some unique problems with historical maps: some features are not present anymore in modern data, sometimes the orthography has changed and historical toponyms might not be in use anymore, all preventing reliable matching of place names.

Method

Our processing pipeline works in five consecutive steps: firstly, image preprocessing to equalise different graphical quality, then detection of text labels with their bounding boxes. Afterwards OCR on said text labels results in actual text strings. In the fourth step, geocoding by querying for recognised text strings thus returns geo-coordinates. Finally transformation and rectification is accomplished with the centers of the bounding boxes and geocoded coordinates as GCP-pairs. These steps will be described in more detail in the following section.

Map preprocessing

As a first step, we apply automatic white balancing by histogram equalisation, to limit the discolouration by paper degradation and allow all further processing steps to work on the full range of colour values.

Maps are commonly printed with a small palette of three to four colours, designed to be easily told apart by the human eye. But when scanning these maps with high resolution and subsequent rasterising, sharp colour transitions get transformed into gradients, resulting in image files with a lot more colours. This makes colour quantisation a non-trivial effort, with the risk of losing too much information, especially on the edges of text, which need to be preserved for successful application of OCR. Three-dimensional k-means clustering on the RGB-space showed unconvincing results because of run-time complexity on large map images and large differences in cluster sizes (e.g. many black pixels, very few blue pixels in many images).

Instead, our approach uses selective gaussian blur to eliminate noise from image compression, smudge hachure lines into a uniform colour and merge the printing dots on very high resolution scans. After an unsharpening filter to revert blurring of edges, a simple but inaccurate colour separation can be obtained by thresholding.

The resulting figure-ground-segmentation remains incomplete, since by colour only some of the features can be removed, while many map elements sharing the same colour as the desired text remain. Examples include roads passing through their designations as well as grid lines and building outlines intersecting place names. Humans have little difficulty separating text from underlying clutter, since their visual perception can rely on Gestalt-principles (Koffka 1935), in particular the law of continuity and the law of past experience, which have yet to be successfully formalised in a computer system.

Some research in text-graphics-separation has been done to tell these conflicting figures, such as text characters and lines, apart on maps, but so far showed unconvincing results even on relatively “clean” datasets, such as computer generated maps from digital vector data (Chiang and Knoblock 2006).

Text Extraction

Notwithstanding the unsatisfactory segmentation, machine learning systems have successfully been trained to detect text in cluttered scenes (Zhou et al. 2017). There has been some research into the peculiarities of detecting text in maps, in particular multi-oriented text in a single document (Roy et al. 2012) and curved text lines (Seytre et al. 2019).

With Strabo, Chiang and Knoblock (2014) have proven the capabilities of their system to robustly and quite accurately deal with aforementioned combined issues effectively to automatically extract the text layer from certain maps and localise individual text labels within. We employ Strabo’s text label detection command-line tool in our experiments without any tuning of parameters or additional training.

The resulting oriented bounding boxes of found text labels in a map are then rotated to be in horizontal orientation and fed to a text-recognition system as individual pages.

We use tesseract² with its python wrapper³, a well-established open source OCR engine, with a multitude of readily-available pre-trained language models. It is necessary to specify the language to be detected, because tesseract’s training makes character detection context sensitive. We use the available pre-trained legacy models, because the newer long short-term memory model does not allow whitelisting of characters, which we deem necessary to avoid erroneous detection of diacritics and special characters in the clutter of overlapping noisy map features.

² <https://github.com/tesseract-ocr>

³ <https://github.com/sirfz/tesseract>

Specifically, we used the pre-trained language model for German and English language without any further additional training and no dictionary for subsequent correction of OCR results.

It is expected to obtain OCR results with only limited quality on maps (Chiang et al. 2016) and it is hardly possible to get accurate transcription without intense manual intervention. But our goal is not to fully digitise the map content: we only require a small number of correctly identified text labels, with a character recognition precision high enough to successfully match these words to a dictionary of place names.

Geocoding

Myers et al. (1996) have already shown the possibility of using gazetteers as dictionaries to be able to find and verify text strings in noisy maps, which can be hard for OCR to correctly recognise with a mere bottom-up method. In place of a dictionary of all place names of potential regions our map might be located in, which can easily contain millions of entries (ca. 2.3 million domestic toponyms are listed by the United States Board on Geographic Names⁴), we employ the web request APIs of OpenStreetMap⁵ (OSM) or GeoNames⁶ for matching and, where possible, localisation of detected place names. Geographical coordinates of identified toponyms obtained by these geocoding services can later be used for automatic georeferencing.

Text labels of feature designations are usually placed somewhat arbitrarily in the vicinity of a point feature (such as a city on large scale maps, or special buildings on small scale maps), along a linear feature (streets or rivers) or within a region (district, geographical area such as landscapes or forest areas or land uses like parks). This makes text labels an inherently imprecise indicator of location. Averaging displacements and disturbances of these location hypotheses over multiple candidate points is required to obtain a good match for the general location of the map.

Generally speaking, different features can be utilised on different scales: small scale maps (1:25.000 and smaller) show street names or point features (churches, plazas), which can lead to very high accuracy, whereas for large scale maps we have to make do with more vague feature designations and rely on averaging out of label displacements and unclear centre points, e.g. of districts. In turn, geodatabases usually store the geographical extent of these larger features, which can serve as a measure of uncertainty for localisation.

There are some special challenges connected to the matching of historical places and toponyms to modern-day databases: some old villages may have been incorporated into cities and only remain as street names. Local place names of regions, like the common name of some valley may not exist as OSM features but may be discovered in street names or an old inn. Sometimes, on the other hand, both the region and a more modern feature exist distinctly but quite close to each other in modern maps. Just as well, former region names might stem from old rivers or brooks, and are now found at multiple different positions along the remaining river run, sometimes many kilometers away. All of these cases lead to multiple hypotheses of locations generated for each detected toponym in a historical map, which will then have to be filtered for the best fitting solution.

⁴ <https://usgs.gov/geonames/domestic-names>

⁵ <https://nominatim.openstreetmap.org>

⁶ <https://www.geonames.org/export/geonames-search.html>

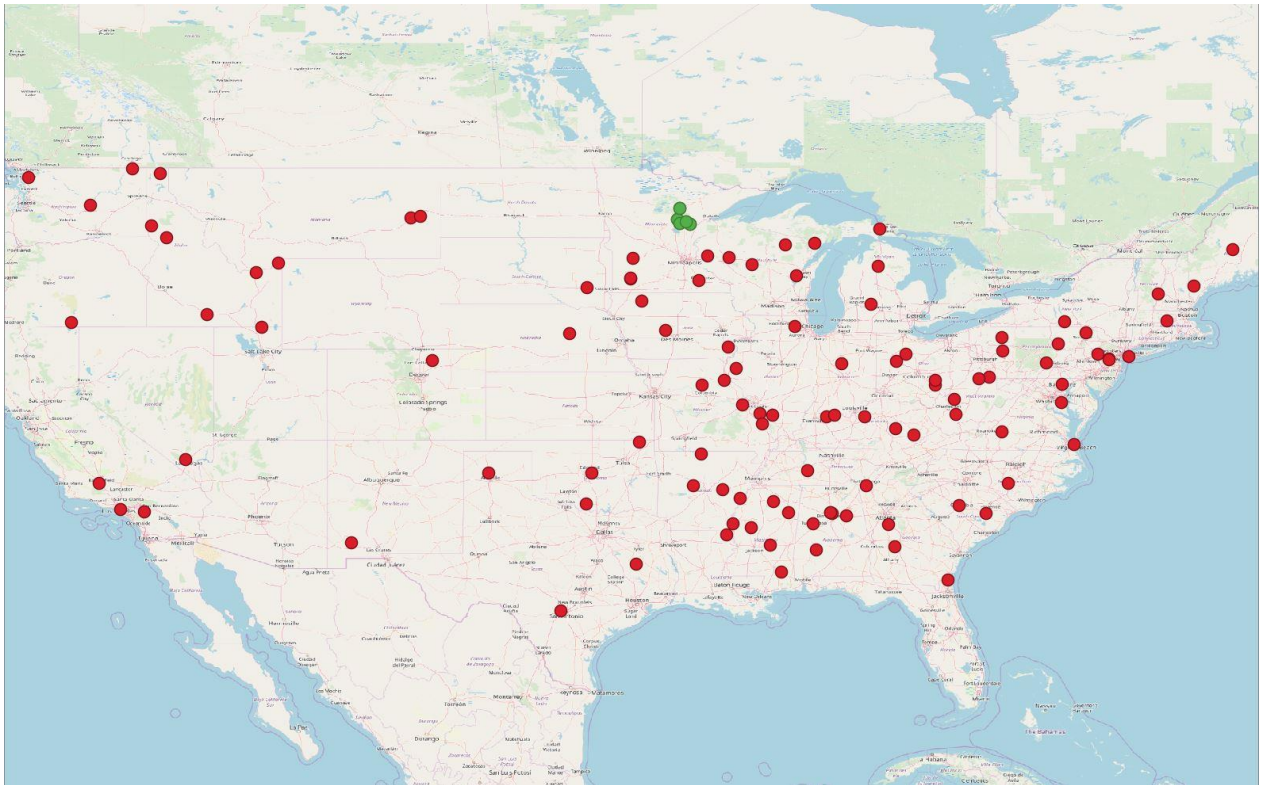


Figure 1. Red: all hypotheses for recognised place names. Green: successful density-based clustering. Background map: OpenStreetMap

Control Point Validation

Especially when recognised designations in a map are not complete because of OCR errors and designations are split over multiple text labels with a commonly used pre- or suffix (e.g. “new...”, “...county”), the place name queries become ambiguous and lead to localisations in far-apart places, even different countries. We can’t rely on the ordering of results, usually by size or importance, of geocoding web services either to help us in correctly identifying, for example, a city with a commonplace name. To illustrate, *Figure 1* shows in red all alternative hypotheses of place locations after geocoding toponyms recognized in a USGS map of Aitkin, Minnesota. In our test dataset of US maps this effect was stronger than in the German maps, possibly stemming from a stronger historical diversification of dialects and thereby toponyms in Europe.

Since even a small number of far off outliers can prohibit successful rectification of a map, we have to establish the correct subset of found hypotheses for further use. We can assume spatial proximity of “correct” localisations (that is coordinates actually representing the specific place in the map), suggesting the use of clustering schemes. Considering the number of outliers can well be in the order of magnitude or even exceed the number of inliers, this leads to inadequate results from common methods like maximum likelihood estimation of Gaussian parameters or random sample consensus.

To overcome this problem, we employ a clustering scheme devised by Ester et al. (1996): density-based spatial clustering of applications with noise (DBSCAN). DBSCAN groups points by determining local neighbourhoods. All points which have a minimum amount of neighbouring points within a specified distance, will be put in a cluster with said neighbours. By the

transitive property, a cluster can extend to neighbours of neighbours to include remote points on the border of the resulting spatial cluster. In our case, location hypotheses obtained by the previous geocoding step are clustered with a minimum distance set to a value smaller than the maximal extent of maps in the dataset, which can be inferred or estimated from the scale or sheet size, which are generally known a priori. As a result, we choose the most promising cluster by the greatest cardinality and use its constituents as GCPs for the following rectification step. The five remaining green dots in *Figure 1* show the effect of said clustering and a successful removal of all outliers.

Transformation

When the map to be transformed uses a known projection and is planimetrically correct, a first-order polynomial or affine transformation should be sufficient for rectification. For this we need at least three valid GCPs. With most maps from the 20th century we don't expect any major distortions within a map sheet and a low order polynomial transformation should give reasonable results. For maps with unusual projections or very large scale, a higher order transformation can help to warp the rectangular map to its correct geographical extents, in turn requiring more GCPs.

Distorted maps need even more complex transformations (e.g. rubbersheeting) which in turn require a lot of accurately placed and evenly distributed GCPs otherwise errors can be amplified in sparsely covered areas.

For our uses, we want a method that can handle some single erroneously predicted locations without losing accuracy at correctly placed GCPs, which lead us to run our experiments with the *thin plate spline* (TPS) transformation.

This more flexible transformation methods is in fact not necessary for the datasets we use, because they are already more or less in the correct orientation and projection, but we use them to test whether our approach can work in general cases and its robustness to misplaced GCPs.

Evaluation

Dataset

We explored the suitability and performance of our approach on the following selection of survey maps: Firstly, German survey maps of Hamburg in 1 : 25 000 from 2002–2004 (TK25_20) in German language latin script including all-capitals, italics and left-leaning italics. They have a palette of three to four colours (black text, buildings, biome pattern symbols, roads, green woodland, blue bodies of water and corresponding name text, brown topography isolines and corresponding height labels. The black text includes districts, hamlets, special land uses, named regions (swamps, state forests, etc.) and some major roads. Another set of maps of overlapping area, but from 1946 to 1959 and previously binarised into low-resolution black-and-white images (TK25_19). Two additional similar colour scanned maps in 1 : 50 000 from 1909 and 1915 were used which only use black colour (TK50).

For comparison we process a selection of USGS historical topographic maps from the years 1953–1985. They use latin script in the English language including all-capitals fonts and have a palette of five colours (red grid lines with corresponding designations, brown topography isolines, green vegetation, blue bodies of water and corresponding text, black administrative boundaries, text, symbols (i.e. mines), roads). The black text designates hamlets, county border

markings, townships, land use (state forests, wildlife preserves, etc.), special point features (lookout towers, airstrips, gravel pits, etc.). Since USGS maps covering rural areas exhibit less clutter, we arbitrarily selected a handful of maps from the American north, mostly Minnesota. We found maps in 1:100.000 scale (USGS100_MN) to give the best results, since smaller scales don't contain many place names for many rural areas of the United States. Conversely, the bigger 1 : 250 000 (USGS250) scale engenders a lot of clutter and overlapping features with comparatively small font sizes.

Some maps use the margins of the map to indicate neighbouring places which lie just outside of the map area, often hinting at where a certain road leads. This writing on the margins, showing little clutter in the background, is happily recognised by the OCR engine, which can lead to significant distortion (see *Figure 2*). The same happens with other writing outside of the map area, such as the map title, issuing authority, or other meta-information. Since there is no easy remedy to this without knowing the exact map extents, we remove the map margins from the test images, in line with a future application of seamless stitching of georeferenced survey map collections.



*Figure 2. Misplaced GCP because of a recognized place name on the map margin.
Transformed Map: TK50 Langenhorn (1909) Background map: OpenStreetMap*

Results

Table 1 shows the results for each processing step for the previously described maps. The second column lists the number of detected labels recovered from Strabo for each sheet in the map. The third column shows the number of labels which were recognised by OCR well enough to return results when querying OSM. Next we list the number of all alternative location hypotheses returned by OSM. In the fourth column the number of GCPs are listed which remain after removing outliers with DBSCAN. Finally, we show from which of these maps a georeferenced output image could be generated.

In total, about 38% of all maps were successfully georeferenced. For some maps enough correct GCPs were placed, but some incorrect ones led to so much distortion, a valid transformation solution could not be determined. Surprisingly, the oldest maps (TK50) delivered the best results (*Figure 5*). Even after 100 years of intense urban development, enough toponyms could be correctly identified to accomplish quite accurate georeferencing. Only a small inset map piece had to be cut out and could not be correctly localised itself.

Table 1. Results comparison for each dataset

Dataset (# of sheets)	# labels	# readable	# after geocoding	# after clustering	successfully georeferenced
TK25_19 (13)	77, 80, 104, 140, 122, 91, 181, 121, 156, 239, 107, 176, 136	4, 6, 5, 8, 3, 6, 19, 5, 12, 14, 5, 2, 5	19, 15, 17, 20, 16, 15, 97, 12, 46, 58, 14, 9, 23,	0, 0, 2, 5, 0, 0, 6, 0, 2, 5, 0, 0, 0	no, no, no, yes, no, no, yes, no, no, yes, no, no, no
TK25_20 (6)	189, 224, 225, 406, 265, 211	25, 13, 7, 19, 18, 24	79, 37, 23, 53, 54, 112	12, 4, 0, 8, 10, 2	yes, yes, no, yes, yes, no
TK50 (3)	83, 70, 117	7, 6, 9	19, 9, 29	3, 3, 2	yes, yes, no
USGS100_MN (4)	119, 82, 133, 46	10, 3, 23, 8	58, 21, 137, 44	0, 0, 5, 0	no, no, yes, no
USGS250 (3)	320, 177, 77	36, 13, 5	180, 83, 22	22, 4, 0	yes, no, no

A good fraction of TK25_20 maps were successfully localised as well. Because of split labels at more complex descriptions like “highway to Oldesloe” or “belonging to Rellingen” some wrongly attributed toponyms lead to distortions. With USGS maps the biggest issues was detecting township or county labels, whereas the smaller settlement labels, often smudged together because of low resolution and colour bleed, could not be recognised and thus geocoding returned wrong locations.

OCR

Experiments show that the label extraction regularly produces too small bounding boxes for the text labels, often cutting half of the first and/or last character of a word (*Figure 3*, left). This seems to happen predominantly with italic fonts. Conversely, sometimes we see too large, ill-fitting bounding boxes with multiple words from different labels, which Chiang et al. (2016) attribute to clutter from non-text symbols in the vicinity of correctly detected text (*Figure 3*, right).

A related, albeit expected, problem is that sometimes designations are split over multiple labels, because of wide spaces between words or a split over multiple lines, which makes it hard to combine them again to the correct place names. When words are split in this way their matches from gazetteers become ambiguous, but the spatial clustering proved to be very helpful in correctly verifying most of these cases.

Overall though, text label detection worked surprisingly well with most of the maps in spite of varying graphical quality. A much greater hindrance is the actual character recognition on the detected labels. This is a strong indicator for inability of the OCR system to deal with map fonts and overlapping features without at least further training. Common OCR engines seem to be trained exclusively on modern typeset fonts and struggle with the writing on maps. As a result we can determine out-of-the-box OCR software does not work satisfactorily on many “technical” stenciled fonts, such as in maps, and will probably be completely unusable on handwritten labels. This is evident from the evaluation of Strabo (Chiang and Knoblock 2014), with a reported 60% to 91% character recall on computer-generated map images from Yahoo and Google respectively, whereas we observe only limited results of far less than 50% recall in our test set of scanned historical maps.

But even with further training on the required fonts, OCR is very sensitive to clutter in figure-colour, especially when overlapping features are touching the characters. This is also the reason why we don't try to detect umlaute and other diacritics, as they are hard to distinguish from the granular scanning noise and some pattern symbols (*Figure 3*, middle). Not detecting diacritical letters is acceptable, since most geocoding services use a fuzzy string matching which works quite well when only supplied with the basic glyph without diacritics. Still, OCR is very sensitive to letter outline defects, so morphological operations to reduce noise and clutter extremely harm recognition precision and should be used with care. In summary, OCR proved to be the biggest bottleneck for successful exploitation of map labels because of difficulties with map-fonts as well as clutter leading to badly recognised characters and words.



Figure 3. Deficiently detected text labels. Taken from map: TK25 Wandsbek (2004)

Registration

While DBSCAN-filtering successfully removes outliers which stem from geocoded location matches of places with similar names, there are still other cases that, while being technically correct matches, still provide erroneous GCPs. Especially in USGS maps the label detection and OCR recognise the names of townships at their centre or of counties along their respective boundary. For the maps in 1 : 250 000 these boundaries can be quite long and thus have been labelled multiple times. When querying these names in geocoding services, they usually return the name of the capital city which gave the name to the respective township or county. This will lead to ground control points, which are more or less in the right area but can distort the output georeferenced map image. Compare *Figure 4*, where OCR recognised township names in a 1 : 100 000 USGS map. The spatial clustering alone can't help with this, instead some semantic handling to deal with these ambiguous designations will have to be investigated in the future.

We settled on mostly using thin plate spline transformation for rectifying map images, because polynomial transformations can shift the entire image to accommodate for minor displacements in the GCPs, which quickly leads to ill fits even at regions around correctly localised points. TPS transformations on the other hand lead to distorted map areas around misplaced GCPs, which might make the map unusable for further applications but is better suitable for understanding the source of distortions. *Figure 4* shows the most problematic property of TPS: when an outlier is on the convex hull of the GCPs, it can quickly shift a big region of the image, even when all other points are correct. Here the top most point stems from the label of a township close to the edge of map, far distant from the actual township center, which is on the next map sheet.

The rectification performance is too bad for actual further use of the maps, but the georeferenced localisation of the map performs actually quite well considering this experimental approach under high uncertainty. To generally handle slightly imprecise GCPs, we just have too few successful localisations. Maybe a dictionary containing generally used terms, such as “marsh”, “lakes”, “forest” or “county”, could be used to remove some of the partially detected

designations, that lead to imprecise localisations. Generally, we need better results from OCR to use contextual information or allow averaging of errors as well as further experiments with transformation methods to elaborate on each methods ability to deal with outliers.



*Figure 4. Misplaced GCPs because of geocoding equating township names and settlement names.
Map: USGS Aitkin (1994).*



Figure 5. Two successfully georeferenced map sheets with used GCPs (red) and ground-truth (blue) Maps: TK50 Langenhorn and Hamburg (1909, 1915)

Future work

The first step for both better recall in label detections as well as improved precision of OCR should be to enhance text-graphics separation, e.g. by symbol and line recognition (Chiang et al. 2006, Heitzler et al. 2018, Velázquez and Levachkine 2003).

Still, for the special use case of historical maps most OCR engines won't work out of the box. We will have to evaluate specialised OCR models or train our own, to obtain sufficient accuracy on stenciled or even handwritten map text. Further training of the Strabo text detector might improve label detection for scanned maps as well.

An additional processing step is called for, using information about differently sized text to infer context information and hierarchy of labels. For example, this could help to determine whether some place names refers to a settlement, county, region or other feature type. Through this better defined search queries can be used while geocoding, to quickly converge on more precise location hypotheses. The same goes for the recurring ambiguous partial place names, resulting from text labels being split over multiple lines or because of wide character spacing. Connecting split text labels (Lin et al. 2018) should greatly facilitate place name matching.

Another problem of geocoding is that the used web searches are very sensitive to single missing or incorrect letters in short queries, despite being able to correct some spelling mistakes. A better fuzzy string matching, that is more lenient to incorrect OCR results could help here, but might require a local copy of the place name database.

The positioning and size of a label alone can already create large geographical displacements. For large scale maps, where not many other features might be present that we can use to obtain more fine-grained alignment, a necessity to improve the accuracy will be to connect the labels to their respective positional markers, if present, as already explored by Budig et al. (2016).

When enough GCPs are left over from the geocoding step we are provided with an over defined solution for finding a transformation function. This can allow us to remove remaining outliers by minimising root-mean-square deviation residuals to increase accuracy from misplaced text labels and wrongly attributed toponyms.

Finally, the use of text labels alone for localisation of GCPs will probably not be precise enough for competing with manual georeferencing. Indeed we only intend it to be a first step of restricting the search space of candidate place matches and a coarse alignment and then further increasing the accuracy by using vectorised features for fine adjustments and stitching together multiple map sheets by other methods as proposed by Briggs and Li (2006) and Chen et al. (2004).

References

- Briggs, R. and Li, Y. (2006). Automated Georeferencing Based on Topological Point Pattern Matching, Proceedings of AutoCarto.
- Budig, B., van Dijk, T. C., Wolff, A. (2016). Matching Labels and Markers in Historical Maps: an Algorithm with Interactive Postprocessing. ACM Transactions on Spatial Algorithms and Systems (TSAS) 2(7). <https://doi.org/10.1145/2994598>
- Chen, I.-W., Chen, H.-R., Tseng, Y.-H. (2016). Automatic image matching and georeferencing of digitized historical aerial photographs. 37th Asian Conference on Remote Sensing (ACRS), 2: 1100–1108.
- Chen, C.-C., Knoblock, C. A., Shahabi, C., Chiang, Y.-Y., Thakkar, S. (2004). Automatically and accurately conflating orthoimagery and street maps. Proceedings of the 12th Annual ACM International Workshop on Geographic Information Systems (ACM-GIS), 47-56.
<https://doi.org/10.1145/1032222.1032231>
- Chiang, Y.-Y., Chiang, Y.-Y., Knoblock, C. A. (2006). Classification of Line and Character Pixels on Raster Maps Using Discrete Cosine Transformation Coefficients and Support Vector Machine. 18th International Conference on Pattern Recognition (ICPR), 2: 1034–1037.
<https://doi.org/10.1109/ICPR.2006.368>
- Chiang, Y.-Y., Knoblock, C. A. (2014). Recognizing text in raster maps. GeoInformatica 19: 1–27. <https://doi.org/10.1007/s10707-014-0203-9>
- Chiang, Y.-Y., Leyk, S., Nazari, N. H., Moghaddam, S., Tan, X. T. (2016). Assessing the impact of graphical quality on automatic text recognition in digital maps, Computers & Geosciences 93: 21–35. <https://doi.org/10.1016/j.cageo.2016.04.013>
- Cléry, I., Pierrot-Deseilligny, M., Vallet, B. (2014). Automatic Georeferencing of a Heritage of old analog aerial Photographs. ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences II–3: 33–40. <https://doi.org/10.5194/isprsannals-II-3-33-2014>

- Ester, M., Kriegel, H.-P., Sander, J., Xu, X. (1996). A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD)*: 226–231.
- Heitzler, M., Gkonos, C., Tsorlini, A., Hurni, L. (2018). A modular process to improve the georeferencing of the Siegfried map. *e-Perimtron*, 13(2): 85–100.
- Koffka, K. (1935). *Principles of gestalt psychology*. London: Kegan Paul, Trench, Trubner And Co.
- Landesamt für Vermessung und Geoinformation Schleswig-Holstein (2004). *Topographische Karte 1:25.000 (TK25), Blatt Wandsbek, Hamburg*.
- Lin, H., Chiang, Y.-Y. (2018). Automatic extraction of phrase-level map labels from historical maps. *SIGSPATIAL Student Research Competition (SRC)*, 9(3): 14–15.
<https://doi.org/10.1145/3178392.3178400>
- Myers, G. K., Mulgaonkar, P. G., Chen, C.-H., DeCurtins, J. L., Chen, E. (1996). Verification-based approach for automated text and feature extraction from raster-scanned maps. *Graphics Recognition Methods and Applications 1072*: 190–203. https://doi.org/10.1007/3-540-61226-2_16
- OpenStreetMap contributors (2020). *OpenStreetMap*. <https://www.openstreetmap.org>
- Roy, P. P., Pal, U., Lladós, J. (2012). Text line extraction in graphical documents using background and foreground information. *International Journal on Document Analysis and Recognition (IJ DAR)* 15(3): 227–241. <https://doi.org/10.1007/s10032-011-0167-3>
- Schröder, J. (2013). Georeferenzierung und Entzerrung der brandenburgischen Kartenblätter des Schmettauschen Kartenwerks. *Vermessung Brandenburg* 2: 14–23.
- Seytre, J., Wu, J., Achille, A. (2019). TextTubes for Detecting Curved Text in the Wild. *ArXiv:1912.08990 [Cs]*.
- United States Geological Survey (1994). 30x60 minute series (topographic), 1:100.000, sheet Aitkin, Minnesota.
- Velázquez, A., & Levachkine, S. (2003). Text/Graphics Separation and Recognition in Raster-Scanned Color Cartographic Maps. *Lecture Notes in Computer Science 3088*: 63–74.
https://doi.org/10.1007/978-3-540-25977-0_6
- Vermessungsbüro der Bau-Deputation Hamburg (1909). *Topographische Karte 1:50.000 (TK50), Blatt Langenhorn, Hamburg*. Amended 1921.
- Vermessungsbüro der Bau-Deputation Hamburg (1915). *Topographische Karte 1:50.000 (TK50), Blatt Hamburg und Amt Ritzebüttel, Hamburg*.
- Wolter, D., Blank, D., & Henrich, A. (2017). Georeferencing River Networks Using Spatial Reasoning. *Proceedings of the 11th Workshop on Geographic Information Retrieval (GIR)*.
<https://doi.org/10.1145/3155902.3155907>
- Zhou, X., Yao, C., Wen, H., Wang, Y., Zhou, S., He, W., Liang, J. (2017). EAST: An Efficient and Accurate Scene Text Detector. *ArXiv:1704.03155 [Cs]*.
<https://doi.org/10.1109/CVPR.2017.283>