

SPRINGER BRIEFS IN GEOGRAPHY

Yao-Yi Chiang · Weiwei Duan
Stefan Leyk · Johannes H. Uhl
Craig A. Knoblock

Using Historical Maps in Scientific Studies Applications, Challenges, and Best Practices



Springer

SpringerBriefs in Geography

SpringerBriefs in Geography presents concise summaries of cutting-edge research and practical applications across the fields of physical, environmental and human geography. It publishes compact refereed monographs under the editorial supervision of an international advisory board with the aim to publish 8 to 12 weeks after acceptance. Volumes are compact, 50 to 125 pages, with a clear focus. The series covers a range of content from professional to academic such as: timely reports of state-of-the art analytical techniques, bridges between new research results, snapshots of hot and/or emerging topics, elaborated thesis, literature reviews, and in-depth case studies.

The scope of the series spans the entire field of geography, with a view to significantly advance research. The character of the series is international and multidisciplinary and will include research areas such as: GIS/cartography, remote sensing, geographical education, geospatial analysis, techniques and modeling, landscape/regional and urban planning, economic geography, housing and the built environment, and quantitative geography. Volumes in this series may analyze past, present and/or future trends, as well as their determinants and consequences. Both solicited and unsolicited manuscripts are considered for publication in this series.

SpringerBriefs in Geography will be of interest to a wide range of individuals with interests in physical, environmental and human geography as well as for researchers from allied disciplines.

More information about this series at <http://www.springer.com/series/10050>

Yao-Yi Chiang • Weiwei Duan • Stefan Leyk
Johannes H. Uhl • Craig A. Knoblock

Using Historical Maps in Scientific Studies

Applications, Challenges, and Best Practices

 Springer

Yao-Yi Chiang
Spatial Sciences Institute
University of Southern California
Los Angeles, CA, USA

Weiwei Duan
Department of Computer Science
University of Southern California
Los Angeles, CA, USA

Stefan Leyk
Department of Geography
University of Colorado
Boulder, CO, USA

Johannes H. Uhl
Department of Geography
University of Colorado
Boulder, CO, USA

Craig A. Knoblock
Information Sciences Institute
University of Southern California
Marina del Rey, CA, USA

ISSN 2211-4165

ISSN 2211-4173 (electronic)

SpringerBriefs in Geography

ISBN 978-3-319-66907-6

ISBN 978-3-319-66908-3 (eBook)

<https://doi.org/10.1007/978-3-319-66908-3>

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2020

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors, and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, express or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

This Springer imprint is published by the registered company Springer Nature Switzerland AG.
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

For Baobao

Acknowledgments

The authors thank Hayley Song, Zekun Li, and Yijun Lin.

This material is based on research supported in part by the National Science Foundation under Grant No. IIS 1563933 (to the University of Colorado at Boulder) and IIS 1564164 (to the University of Southern California), in part by the National Endowment for the Humanities (Grant No. NEH PR-253386-17), in part by the University of Southern California Undergraduate Research Associates Program (USC URAP), and in part by Microsoft and NVIDIA Corporation.

Contents

1	Introduction	1
1.1	Book Objectives	1
1.2	Book Structure	7
	References	7
2	Historical Map Applications and Processing Technologies	9
2.1	Introduction	10
2.2	Applications of Historical Maps	13
2.3	Case Studies of Map Processing Technologies	19
2.3.1	Case Study I: Semi-Automatic Symbol Recognition from Map Scans	21
2.3.2	Case Study II: Multi-Model, Context-Based Automatic Symbol Recognition from Map Scans	29
2.3.3	Case Study Discussion and Outlook	32
2.4	Chapter Summary	33
	References	34
3	Creating Structured, Linked Geographic Data from Historical Maps: Challenges and Trends	37
3.1	Introduction	37
3.2	Finding Relevant Historical Maps	43
3.3	Converting Map Content to Machine-Readable Formats and Record Uncertainty	46
3.3.1	Crowdsourcing Approaches	47
3.3.2	Semi-automatic Approaches	49
3.3.3	Multi-Model, Context-Based, Automatic Approaches	51
3.4	Modeling and Publishing Map Content	55
3.5	Chapter Summary	59
	References	60

4 Training Deep Learning Models for Geographic Feature Recognition from Historical Maps	65
4.1 Introduction	66
4.2 Challenges in Using CNNs on Historical Maps	67
4.2.1 Accurate Boundary Delineation of Geographic Features	69
4.2.2 Scarce Training Data for Cartographic Documents	70
4.3 Overview of Semantic Segmentation for Geographic Feature Recognition from Map Scans	74
4.3.1 VGG16: The 16-layer Very Deep Convolutional Networks for Large-Scale Image Recognition	75
4.3.2 GoogLeNet	77
4.3.3 ResNet	78
4.3.4 The Encoder and Decoder Architecture for Semantic Segmentation	79
4.3.5 Multi-Scale Pyramids of Feature Images for Semantic Segmentation	82
4.4 Overview of Transfer Learning for Geographic Feature Recognition from Map Scans	84
4.5 Experiment	85
4.5.1 Experimental Data, Settings and Evaluation Metrics	86
4.5.2 Experiment I: The Impact of Backbone CNNs: FCN-VGG16, FCN-GoogLeNet, and FCN-ResNet	88
4.5.3 Experiment II: The Impact of Transfer Learning Strategies: PSPNet	89
4.6 Chapter Summary	96
References	96
5 Summary and Discussion	99
5.1 Book Summary	99
A Railroad Recognition Results	101

Chapter 1

Introduction



Abstract Historical maps are fascinating to look at and contain valuable retrospective place information difficult to find elsewhere. However, the full potential of historical maps has not been realized because the users of scanned historical maps and the developers of digital map processing technologies are from a wide range of disciplines and often work in silos. This book aims to make the first connection between the map user community and the developers of digital map processing technologies by illustrating several applications, challenges, and best practices in working with historical maps. This chapter presents a brief introduction to various types of historical maps and the scientific studies that could benefit from using them. Further, the chapter summarizes the general considerations critical for building successful computational processes that can be used to analyze historical map content. Finally, the chapter provides an overview of the book structure, describing the connections between individual chapters.

1.1 Book Objectives

Ever since the prehistorical times, various forms of maps represent an important medium for visual communication about human-made elements, human activities, and natural phenomena on Earth (or in the sky) as well as their spatial relationships.¹ Between cave paintings to Google Maps, the idea of mapmaking has gone from flat Earth to numerous spatial reference systems, and the purpose of maps has progressed from storytelling to navigation. During the period of hundreds and thousands of years, mapmakers (cartographers) have produced a significant amount of maps. For example, in 1502, Leonardo da Vinci created the Imola town plan (Fig. 1.1) [BF07], which is arguably the first scientifically measured map (or ichnographic) [BL14]. Although produced with simple measuring tools by today's standard, the da Vinci's Imola map still mostly matches the current street layout

¹The reader is referred to the multi-volume book [HWL87] for a grand overview of map making (cartography) from prehistorical times to the twentieth century.



Fig. 1.1 Town plan of Imola, Italy, Leonardo da Vinci (circa 1502) (source: https://upload.wikimedia.org/wikipedia/commons/d/d1/Leonardo_da_vinci%2C_Town_plan_of_Imola.jpg)

in Imola after more than 500 years. This map of Imola and many other similar historical city plans open a window for current day researchers to look at cities in the distant past while special considerations of epistemological changes [Har87] and metric accuracy [BL14] must be taken into account.

Fast forward to the twentieth century, more recent historical maps were created with modern technologies such as geodetic measuring technologies, satellite imagery and aerial photos (as a reference source), and even the Global Positioning System (GPS), but these maps still mostly exist only in paper format.² For example, during the World War II (WWII), the Great Britain War Office and United States Army Map Service (AMS) together created numerous paper maps covering a large amount of Earth surface with abundant retrospective information about places. Other map publishers, such as the Polish Centrum Kartografii, also offer pre- and post-WWII maps (in paper format) of Europe with a comprehensive list of retrospective place names including towns, monuments, etc. In the USA, the US Geological Survey (USGS) historical topographic maps (1884–2006) contain professionally surveyed geographic features of detailed natural and manmade objects,

²Professional cartographers would engrave map features on several metal plates and then combine them to print the maps with each plate using an appropriate color ink (e.g., brown for topographic contour lines and black for text labels and roads.)



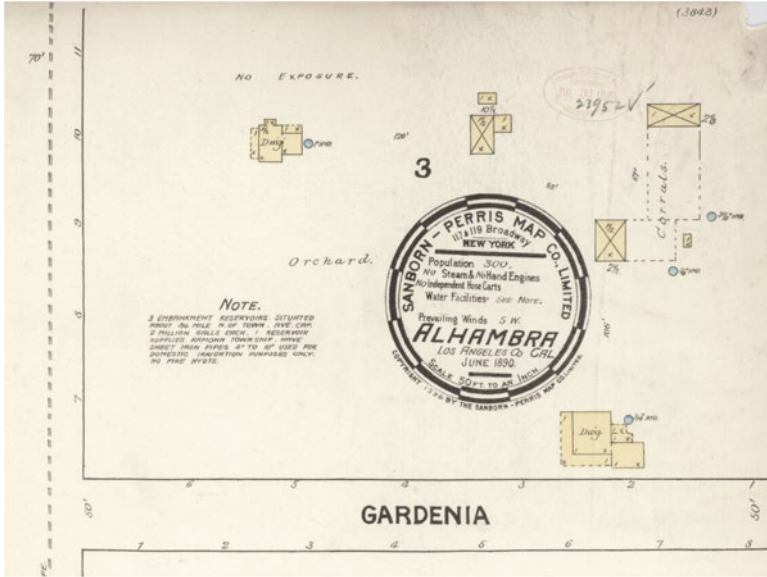
(a)



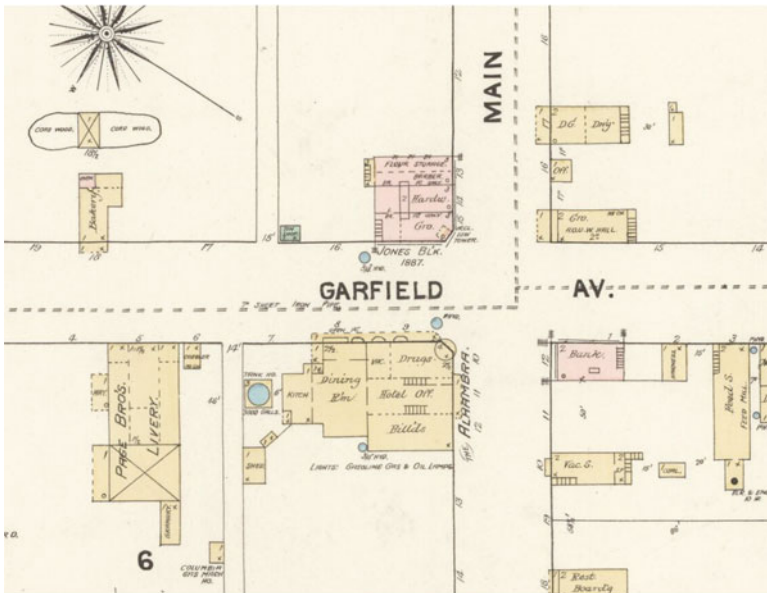
(b)

Fig. 1.2 Historical USGS topographic maps: (top) Los Angeles, CA (1:24K) (circa 1931), (bottom) San Bernardino, CA (1:250K) (circa 1966). (a) Roads and buildings (black). (b) Urban areas (orange), roads (red), railroads (black)

such as mountains, lakes, rivers, vegetation, buildings, transportation networks, and place names as well as boundaries of political and urban areas (Fig. 1.2). These recent maps can contain very detailed retrospective place information about the past landscape. As another example, Fig. 1.3 shows two sample sections of the Sanborn Fire Insurance Map archived in the United States Library of Congress. The Sanborn maps provide detailed retrospective place information such as the prevailing wind speed and direction of the city, business locations and types, building outlines, and building materials.



(a)



(b)

Fig. 1.3 Sanborn Fire Insurance Map from Alhambra, Los Angeles County, California (source: https://www.loc.gov/resource/g4364am.g4364am_g003761890/). (a) Information about the city population, water facilities, and prevailing winds (direction and speed). (b) Detailed building information, including business types and materials (red: bricks, brown: adobe), etc.

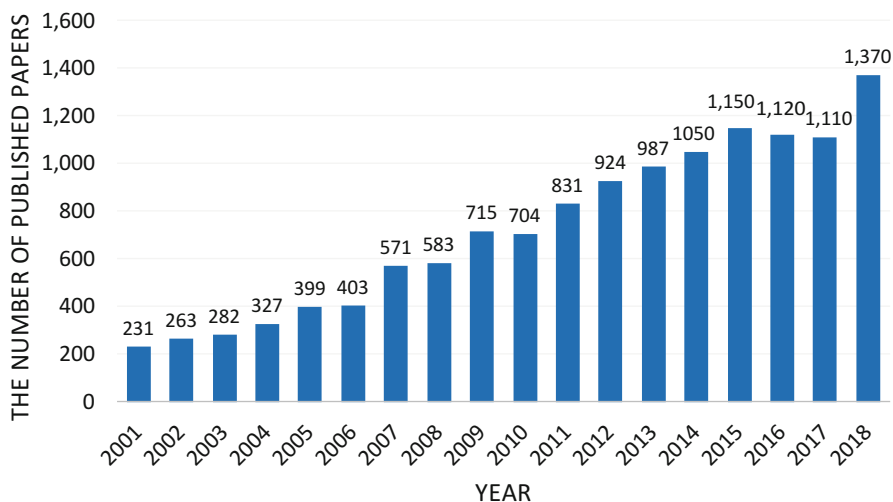


Fig. 1.4 The number of published papers per year using the search keywords “historical maps,” “change” and “analysis” on Google Scholar

Today, online map archives and libraries such as the David Rumsey Map Collection³ and the United States Library of Congress’s digital map collection⁴ store a large number of historical maps in either paper or scanned format. These historical maps have the potential to unlock unique research opportunities in various disciplines in the social and natural sciences. For example, historical map series such as the USGS historical topographic maps⁵ provide thousands of maps with detailed retrospective place information covering large, national scale geographic areas and long time periods. Figure 1.4 illustrates the broad potential for scientific studies using historical map data: searching Google Scholar using the keywords “historical maps,” “change,” and “analysis” returns about 15,700 results of which around 13,020 appeared after 2001. These results include top-ranked scientific journals such as *Nature*, *Remote Sensing of Environment*, *Ecology*, *Global Change Biology*, *Landscape and Urban Planning*, *Landscape Ecology*, *Land Use Policy*, *Journal of Glaciology*, *Environment and Planning B*, *Forest Ecology and Management*, *Agriculture*, *Applied Geography*, and *Journal of Coastal Research*. Since geographic (or place) information is an important component that links heterogeneous data used in many academic disciplines and application domains [Jan+12, Get+11], detailed information about places and their changes is crucial for studies in sociology (e.g., [Kur13]), cancer and environmental epidemiology (e.g., [Mac04]), urbanization, biodiversity (e.g., Hill et al. [Hil+09]), human disease (e.g., [Yos+14]),

³<http://www.davidrumsey.com/>.

⁴<https://www.loc.gov/collections/?fa=original-format:map>.

⁵<http://ngmdb.usgs.gov/maps/TopoView/>.

and biology (e.g., [Dav+15, Lav13, Vel+13]). These recent studies illustrate the vast diversity of scientific investigations using historical maps, indicating that a broad interdisciplinary research community will benefit from harnessing massive and heterogeneous map data to enable new forms of data-intensive research. Also, the tools that allow efficiently and effectively converting large amounts of historical map scans into structured databases will help researchers and map curators to transform their map collections into a useful format for geospatial analysis.

While the computer science and geographic information science community has been developing technologies for automatic and semi-automatic map understanding (digital map processing) for almost 40 years [CLK14], currently the mainstream method for making historical maps useful in scientific studies still relies on manual digitization with very little help from intelligent algorithms or systems. Therefore, the potential of historical maps has not been fully realized. For example, most land-change analyses involving pre-1970 data⁶ often rely on manual map digitization and thus are constrained to study areas covered by very few maps [Kie93, PL02, Die+05, KET07, Ste+10]. Although many historical maps are now in digital format as scanned images, systematic exploration of their contents requires robust, efficient digital map processing technologies for the extraction and conversion of image data into a format that allows meaningful analysis in a geographic information system (GIS). Also, these computational technologies need to be developed collaboratively with the users of historical maps so that the users can easily adopt new technologies and use them in their work.⁷ Because of the disconnection between the users and developers of digital map processing technologies, very often, digital map processing is thought of as a black box that can only process specific types of map images and require advanced knowledge in image processing and computer vision to operate. This unfortunate disconnect could be a result of a dispersed digital map processing community within computer science and geographic information science and the diverse user communities spread across a broad range of disciplines.

This book intends to provide an introduction of the current applications, challenges, and best practices of digital map processing technologies aiming to illustrate what to expect in dealing with historical maps using computational methods for both the user of historical map content and the developer of digital map processing technologies. The goal is to bring together the communities that have an implicit interest in historical maps with the hope of making widely available the valuable content that is difficult to find anywhere else but remains “locked” in cartographic documents.

⁶The first Landsat satellite was operational in the 1970s.

⁷For example, a map processing system for text recognition requiring the user to first collect thousands of text samples from their maps for training is impractical for a scientist who wants to process two map scans from the sixteenth century.

1.2 Book Structure

The remainder of this book is structured as follows: Chapter 2 first presents a variety of existing applications of historical maps to demonstrate varying needs for processing historical maps in scientific studies (e.g., thousands of historical maps from a map series vs. a few historical maps from various publishers and with different cartographic styles). Chapter 2 also describes case studies introducing typical types of semi-automatic and automatic digital map processing technologies. The case studies showcase the strengths and weaknesses of semi-automatic and automatic approaches by testing them in a symbol recognition task on the same scanned map.

Chapter 3 presents the technical challenges and trends in building a map processing, modeling, linking, and publishing framework. The framework will enable querying historical map collections as a unified and structured spatiotemporal source in which individual geographic phenomena (extracted from maps) are modeled (described) with semantic descriptions and linked to other data sources (e.g., DBpedia, a structured version of Wikipedia).⁸

Chapter 4 dives into the recent advancement in deep learning technologies and their applications on digital map processing. The chapter reviews existing deep learning models for their capabilities on geographic feature extraction from historical maps and compares different types of training strategies. A comprehensive experiment is described to compare different models and their performance.

Chapter 5 closes the book with a brief summary.

Each chapter can be read individually, but the order of chapters in this book helps the reader to first understand the “product requirements” of a successful digital map processing system, then review the existing challenges and technologies, and finally follow the more recent trend of deep learning applications for processing historical maps.

References

- [BF07] H. Ballon, D.H. Friedman, Portraying the city in early modern Europe: measurement, representation, and planning, in *The History of Cartography* (University of Chicago Press, Chicago, 2007), pp. 680–704
- [BL14] V. Baiocchi, K. Lelo, Assessing the accuracy of historical maps of cities: methods and problems. *Citta e Storia* 9(1), 61–89 (2014)
- [CLK14] Y.-Y. Chiang, S. Leyk, C.A. Knoblock, A survey of digital map processing techniques. *ACM Comput. Surv.* 47(1), 1–44 (2014). <https://doi.org/10.1145/2557423>. ISSN: 0360-0300
- [Dav+15] C.C. Davis, C.G. Willis, B. Connolly, C. Kelly, A.M. Ellison, Herbarium records are reliable sources of phenological change driven by climate and provide novel insights

⁸<https://wiki.dbpedia.org>.

- into species' phenological cueing mechanisms. *Am. J. Bot.* **102**(10), 1599–1609 (2015). <https://doi.org/10.3732/ajb.1500237>. ISSN: 0002-9122, 1537-2197
- [Die+05] C. Dietzel, M. Herold, J.J. Hemphill, K.C. Clarke, Spatiotemporal dynamics in California's central valley: empirical links to urban theory. *Int. J. Geogr. Inf. Sci.* **19**(2), 175–195 (2005). <https://doi.org/10.1080/13658810410001713407>
- [Get+11] A. Getis, M. Goodchild, N. Lam, C. Merry, T. Nyerges, H. Onsrud, B. Plewe, K. Stewart, L. Usery, A framework for the future of the spatial sciences, in *2011 Workshop on the Future of the Spatial Sciences* (2011)
- [Har87] J.B. Harley, The map and the development of the history of cartography, in *The History of Cartography. Volume 1: Cartography in Prehistoric, Ancient, and Medieval Europe and the Mediterranean* (University of Chicago Press, Chicago, 1987)
- [Hil+09] A.W. Hill, R. Guralnick, P. Flemons, R. Beaman, J. Wieczorek, A. Ranipeta, V. Chavan, D. Remsen, Location, location, location: utilizing pipelines and services to more effectively georeference the world's biodiversity data. *BMC Bioinform.* **10**(suppl 14), S3 (2009). <https://doi.org/10.1186/1471-2105-10-S14-S3>. ISSN: 1471-2105
- [HWL87] J.B. Harley, D. Woodward, G.M. Lewis, *The History of Cartography*, vol. 1 (University of Chicago Press, Chicago, 1987)
- [Jan+12] K. Janowicz, S. Scheider, T. Pehle, G. Hart, Geospatial semantics and linked spatiotemporal data – past, present, and future. *Semantic Web* **34**, 321–332 (2012). <https://doi.org/10.3233/SW-2012-0077>
- [KET07] J. Kozak, C. Estreguil, M. Troll, Forest cover changes in the northern Carpathians in the 20th century: a slow transition. *J. Land Use Sci.* **2**, 127–146 (2007). <https://doi.org/10.1080/17474230701218244>
- [Kie93] F. Kienast, Analysis of historic landscape patterns with a Geographical Information System — a methodological outline. *Landsc. Ecol.* **8**(2), 103–118 (1993). <https://doi.org/10.1007/BF00141590>. ISSN: 1572-9761
- [Kur13] L. Kurashige, Rethinking anti-immigrant racism: lessons from the Los Angeles vote on the 1920 Alien land law. *Southern California Quart.* **95**(3), 265–283 (2013). <https://doi.org/10.1525/scq.2013.95.3.265>. ISSN: 0038-3929
- [Lav13] C. Lavoie, Biological collections in an ever changing world: herbaria as tools for biogeographical and environmental studies. *Perspect. Plant Ecol. Evol. Systemat.* **15**(1), 68–76 (2013). <https://doi.org/10.1016/j.ppees.2012.10.002>. ISSN: 1433-8319
- [Mac04] T.M. Mack, *Cancers in the Urban Environment* (Elsevier Science, Amsterdam, 2004). ISBN: 9780080528465
- [PL02] C. Petit, E. Lambin, Impact of data integration technique on historical land-use/land-cover change: comparing historical maps with remote sensing data in the Belgian Ardennes. *Landsc. Ecol.* **172**, 117–132 (2002). <https://doi.org/10.1023/A:1016599627798>. ISSN: 1572-9761
- [Ste+10] E.D. Stein, S. Dark, T. Longcore, R. Grossinger, N. Hall, M. Beland, Historical ecology as a tool for assessing landscape change and informing wetland restoration priorities. *Wetlands* **30**(3), 589–601 (2010)
- [Vel+13] M. Vellend, C.D. Brown, H.M. Kharouba, J.L. McCune, I.H. Myers-Smith, Historical ecology: using unconventional data sources to test for effects of global environmental change. *Am. J. Bot.* **100**(7), 1294–1305 (2013). <https://doi.org/10.3732/ajb.1200503>. ISSN: 0002-9122, 1537–2197
- [Yos+14] K. Yoshida, H.A. Burbano, J. Krause, M. Thines, D. Weigel, S. Kamoun, Mining herbaria for plant pathogen genomes: back to the future. *PLoS Pathog.* **10**(4), e1004028 (2014). <https://doi.org/10.1371/journal.ppat.1004028>. ISSN: 1553-7366, 1553-7374

Chapter 2

Historical Map Applications and Processing Technologies



Abstract Digital map processing has been an interest in the computer science and geographic information science communities since the early 1980s. With the increase of available map scans, a variety of researchers in the natural and social sciences developed a growing interest in using historical maps in their studies. The lack of an understanding of how historical maps can be used in research and the capabilities of map processing technologies creates a significant gap between the wide range of communities that could benefit from the advances in digital map processing technologies and the disciplines in which the technologies are developed. As a result, researchers who intend to use historical maps in their studies still need a significant amount of resources to digitize their maps, while the existing digital map processing technologies are difficult to apply and understand and thus do not look promising. In many cases, existing digital map processing technologies could help facilitate the digitization process, and it just requires additional knowledge to select an appropriate technology given the problem scope (e.g., the number of maps for processing, map conditions, and style varieties). The result is that researchers waste time and resources building and testing various systems that partially duplicate prior work and cannot fully use the potential of existing technology. This chapter presents real-world applications of historical maps and case studies of both semi-automatic and fully automatic approaches for geographic feature extraction from historical maps. These real-world applications illustrate and exemplify various needs and scopes of using historical maps in scientific studies (e.g., processing thousands of historical maps from a map series vs. a few historical maps from various publishers and with different cartographic styles). The two example map processing technologies described help understand current strengths and weaknesses. These examples also illustrate tremendous collaboration opportunities between and beyond the computer science and geographic information science communities to build advanced map processing technologies that are more effective in transforming the scientific studies that use historical maps.

2.1 Introduction

Historical maps are an irreplaceable *primary source* of geographical and political information in the past (e.g., historical place names, landmarks, natural features, transportation networks, and war, trade, and diplomacy networks). For example, the Mappa Mundi by Fra Mauro (circa 1450) (Fig. 2.1) contains not only place names but also provides

... natural philosophy, description of places and people, commercial geography, history, navigation and direction of expansion, and, finally, on what we can nowadays call methodological issues. In addition, Fra Mauro's world map also includes hundreds of images, representing cities, temples, funerary monuments, streets, and ships, as well as a scene in the lower left corner representing Earthly Paradise. [Nan+15, pg. 160]



Fig. 2.1 East Asia Mainland of the Mappa Mundi (circa 1450), Fra Mauro

In many cases, historical maps are also the only source that provides professionally surveyed historical geographic data. Map archives such as the US Geological Survey (USGS) National Geologic Map Database,¹ USGS Topographic Maps,² David Rumsey Map Collection,³ OldMapsOnline.org,⁴ and the National Library of Scotland⁵ together store millions of this type of historical map in either paper or scanned formats. For example, between 1884 and 2006, the USGS has created over 200,000 topographic maps. According to the USGS, in the USA, these topographic maps

... portray both natural and manmade features. These maps show and name works of nature including mountains, valleys, plains, lakes, rivers, and vegetation. They also identify the principal works of man, such as roads, boundaries, transmission lines, and major buildings.⁶

The USGS National Geospatial Program has scanned these historical paper maps. Collectively, these publicly available scanned maps portray the evolution of the American landscape over a 125-year period. Similar map series exist in many countries, e.g., the Ordnance Survey maps in the UK archived by the National Library of Scotland. In this case of more recent historical maps produced using modern geospatial survey technologies (e.g., the USGS Topographic Map series, Ordnance Survey six-inch series, and other national agency series dated from the early 1800), the detailed map data on the states of landscapes in the past are essential for understanding the causes and consequences of environmental change and support a variety of natural and social studies on topics such as cancer and environmental epidemiology, urbanization, biodiversity, landscape changes, and history.⁷ However, many of these historical maps are not georeferenced, and almost all historical maps have content that is not machine-readable.

With the exponential growth of available map scans stored and maintained in digital archives and on the Internet, a variety of disciplines in the natural and social sciences developed a growing interest in using historical maps in their studies. To make historical maps usable in an analytic environment (e.g., a Geographic Information System, GIS), the computer science and geographic information science communities have been developing computational methods for the extraction and recognition of the content from images of maps since the early 1980s [CLK14], but the existing map processing technologies are still limited in processing a large number and variety of map images.⁸ Today, a researcher can spend a great deal of time and effort searching and cross-referencing data

¹http://ngmdb.usgs.gov/ngmdb/ngmdb_home.html.

²<http://ngmdb.usgs.gov/maps/TopoView/>.

³<http://www.davidrumsey.com/>.

⁴<http://www.oldmapsonline.org/>.

⁵<http://maps.nls.uk/>.

⁶<https://pubs.usgs.gov/gip/topomapping/topo.html>.

⁷The reader is referred to [Gre+15] for more examples and methodologies in historical geographic information systems (GISs).

⁸The reader is referred to [CLK14] and [Chi+16] for detailed reviews on map processing techniques and systems.

sources to find relevant maps. Then they need to select the most efficient and effective method to digitize the maps for converting the map content to a machine-readable format (e.g., [GE15, Nan+15]). The researcher may need to search in various data repositories, map repositories, search engines, and even when they have identified the map(s), it just requires additional knowledge to select an appropriate technology for map digitization given the problem scope (e.g., the number of maps for processing, map conditions and varieties). Consequently, the mainstream method for map digitization is still manual digitization using a GIS or image processing software. These challenges in working with historical maps present an enormous collaboration opportunity for the computer science and geographic information science communities to build advanced map processing technologies for transforming the scientific studies that currently use limited content from only a small percentage of available historical maps. Therefore, it is important to understand the current landscape of existing and potential applications of historical maps and the types of technologies that can be used to support these applications.

This chapter presents real-world applications of historical maps in scientific studies and case studies of semi-automatic and fully automatic geographic feature extraction from historical maps, aiming to bridge the gap between the users and developers of map processing technologies. This chapter first presents the potentials and real-world applications of historical maps in a variety of studies, including topics in natural and social sciences. These applications demonstrate various types of requirements for historical maps in scientific studies. For example, some studies require processing only a few maps, but these maps can come from multiple sources and have very different cartographic styles. This type of work can take advantages of semi-automatic map processing algorithms and systems that need some user intervention steps but can process a wide variety of maps. Other studies might need to process hundreds of historical maps from the same map series in an archive to understand the long-term changes in the landscape. This type of work can use automatic processing algorithms and systems that can be trained in a one-time training effort using appropriate training data. Following on the discussion of historical map applications, this chapter then presents two case studies on a similar application to demonstrate the strengths and weaknesses of typical types of semi-automatic and fully automatic map processing technologies.

The remainder of this chapter⁹ is structured as follows: Section 2.2 describes the potentials and real-world applications of historical maps in a variety of studies, including topics in the natural and social sciences. Section 2.3.1 presents a case study on semi-automatic technologies for extracting geographic features from historical maps. Semi-automatic technologies require some user intervention steps but are often more robust than fully automatic map processing systems. Section 2.3.2 presents a second case study on extracting geographic features from historical

⁹This chapter is based on the materials in [Chi17b] with major extensions of case studies on geographic feature extraction applications.

maps with context-based, fully automatic technologies. Section 2.4 summarizes the chapter and discusses future work.

2.2 Applications of Historical Maps

Historical data archives (e.g., museum and herbaria collections, digital photography and newspaper archives) support a variety of scientific studies in natural science on topics such as biodiversity (e.g., [Hil+09]), evolutionary biology (e.g., [Lav13]), human disease (e.g., [Yos+14]), plant biology [Dav+15, Vel+13], and ecology (e.g., [New10, PE10]), but geolocating the historical localities mentioned in archives (e.g., Calflora Observation Database¹⁰ and the Global Biodiversity Information Facility [Sam+13]) is challenging and very often a tedious manual process using historical maps. Murphey et al. [Mur+04] reviewed the problems in georeferencing museum collections. They compared several geoparsing tools including the GEOLocate [RB10] and BioGeomancer [Gur+06] for converting text descriptions to locations. Since then, a variety of advanced algorithms for geoparsing has been proposed (e.g., [LL11]) and open-source software packages (e.g., CLAVIN,¹¹ CLIFF [DBZ], and the Edinburgh Geoparser [Ale+15]) are available. These algorithms and tools are widely used in geolocating places in the unstructured text and used in spatial humanities research (e.g., [Gre+15]). However, these tools need a “gold data” gazetteer to provide the location information of recognized place names, and the lack of historical reference gazetteers remains a challenge. The result is that even if the geoparsing software can correctly identify a historical name as a geolocation reference in the unstructured text, the geocoordinates of the historical name is still unknown if the place name no longer exists.

To locate the place names that no longer exist in contemporary data sources, a researcher needs to search and cross-reference a variety of data sources such as archives of historical maps, newspapers, and photography. For example, a data record in an online database of California herbarium specimens describes an August 16th, 1902 observation of *Artemisia douglasiana* (California mugwort) at the location “near Mesmer” in Los Angeles. The place name Mesmer near or within both the City and County of Los Angeles no longer exists in the contemporary geographic data sources, including authoritative sources like the US Census¹² and USGS GNIS (the United States Geological Survey Geographic Names Information System)¹³ and open sources, such as GeoNames,¹⁴ OpenStreetMap,¹⁵

¹⁰<http://www.calflora.org/>.

¹¹<https://clavin.bericotechnologies.com/>.

¹²<https://www.census.gov/geo/maps-data/data/gazetteer.html>.

¹³<http://geonames.usgs.gov/>.

¹⁴<http://www.geonames.org/>.

¹⁵<https://www.openstreetmap.org/>.



Fig. 2.2 Map of the Mesmer city development

and Wikipedia. Searching “Mesmer” in the GeoNames gazetteer results in an airport “Mesmer Airport” in New York and a street “Rue Mesmer” in Haiti. Neither of the results helps to geolocate the observation of California mugwort in 1902. A Google search with the keywords “Mesmer” and “Los Angeles” reveals a few interesting facts that could be helpful for geolocating Mesmer. First, the search results include a person, Louis Mesmer (1829–1900), who was a prominent businessman and the owner of the famous US Hotel in Los Angeles. Because it was common to name locations after well-known families (e.g., Wilshire, Hancock, and Doheny in Southern California), Mesmer could be a place name in the Los Angeles area in the past. Second, the search results contain a link to a map in the Los Angeles Public Library collections showing a proposed development plan in 1924 for the “Mesmer City” in Los Angeles (Fig. 2.2). At the time, Mesmer City was advertised as “*In the direct path of the Los Angeles’ growth toward the ocean.*”¹⁶ This map further narrows down the search space for Mesmer to somewhere nearby Culver City and Baldwin Hills in Los Angeles. Together, the time and location information from the search results points to the USGS topographic map that contains the Mesmer in 1901 (Fig. 2.3). In this case, Mesmer is geolocated, but the entire process cannot scale to handle thousands of records in an efficient manner.

Historical GIS (Geographic Information System) [GE07] could alleviate the problem of geolocating historical locality references by providing a platform for collecting datasets of historical place names, but the datasets are rarely available. Even when historical gazetteers are available, their spatiotemporal coverage is often sparse. For example, the US Census only provides post-2010 and 2000 and 1990 census gazetteer files. NHGIS (the National Historical Geographic Information System at the Minnesota Population Center [Man+17])¹⁷ provides historical demography data down to the census tract level but only a few place names. The Ramsay Place Names File from the State Historical Society of Missouri provides a historical

¹⁶<https://www.lapl.org/collections-resources/visual-collections/map-collection/>.

¹⁷<https://www.nhgis.org/>.



Fig. 2.3 The USGS historical topographic map shows the location of Mesmer. (Southern California sheet No. 1, circa 1901)

gazetteer covering locations in the State of Missouri from 1928 to 1945 [Ada28]. An exceptional case is the website “A Vision of Britain through Time”¹⁸ from the GB Historical GIS at the University of Portsmouth, which provides about two million crowdsourced historical place names and other locality text in Great Britain dated back in the early nineteenth century from the Ordnance Survey historical six-inch map series (second edition, county series).

The above examples of historical inquiries show that the ability to automatically use the textual content from a large number of historical maps as the locality reference source (e.g., two million crowdsourced place names and locality text from approximately 8000 maps) will be able to transform historical data records in documents and collections into georeferenced datasets. This ability will enable natural and social science researchers to efficiently find, query, and analyze a variety of historical records by location.

As another example of the use of historical maps in scientific studies, Kurashige [Kur13]¹⁹ used historical census data, voting records, and precinct numbers and boundaries extracted from a 1920 map to study “who” (e.g., occupations and political parties) in Los Angeles voted for the 1920 California Alien Land Law that discriminates against Japanese (Fig. 2.4).

Ngo et al. [NSC15] built a Web-based interactive visualization tool for historical land reclamation records in Hong Kong.²⁰ Ngo collected and manually digitized

¹⁸<http://www.visionofbritain.org.uk/>.

¹⁹Dr. Kurashige’s article published in the *Southern California Quarterly* won the 2015 Carl I. Wheat Award for the best demonstration of scholarship in that journal from 2012–2014 by a senior historian.

²⁰This Web tool is among the top hits when searching for Hong Kong land reclamation on the Google search engine: <http://www.oldhkphoto.com/coast/>.

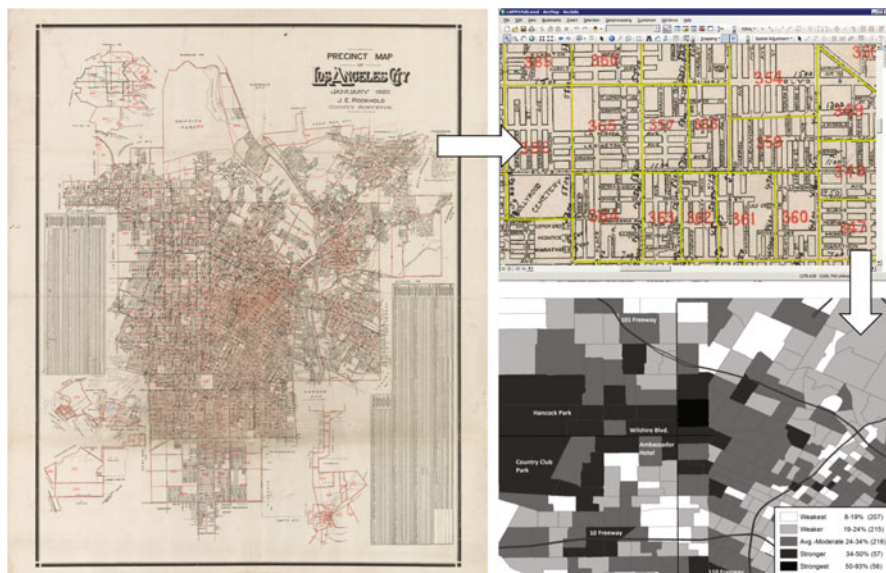


Fig. 2.4 Automatically extracting precinct boundaries from a historical precinct map of Los Angeles for analyzing historical voting records with demographic datasets (Figure adapted from [Kur13])

(using Google Maps) several historical maps and land records, which were then cross-referenced to create both the geometry and attributes (e.g., the year of land reclamation) of the historical land reclamation records in Hong Kong (Fig. 2.5). The historical maps and land records are from the Hong Kong Public Library (Map Library Collection),²¹ the University of Hong Kong Libraries (Special Collection),²² and the Hong Kong Lands Department.²³ Figure 2.6 shows a sample of the digitization results and the variety of historical maps used to recreate the land reclamation records. The digitization results were also used to compile infographics and maps to demonstrate coastline changes in Hong Kong [Yu14].

The Spatial Sciences Institute at the University of Southern California (USC) collaborated with an insurance company to automatically read historical Ordnance Survey maps (circa 1900–1970) covering the entire UK to identify likely locations of subterranean contamination, such as factories, mines, quarries, and gas works which no longer exist and otherwise would not be known today (Fig. 2.7).²⁴

²¹ <https://www.hkpl.gov.hk/en/about-us/HKCL/services/map.html>.

²² <https://lib.hku.hk/hkspc/>.

²³ <http://www.landsd.gov.hk/mapping/en/download/maps.htm>.

²⁴ Spatial technology opens a window into history: <https://news.usc.edu/91625/spatial-technology-opens-a-window-into-history/>.

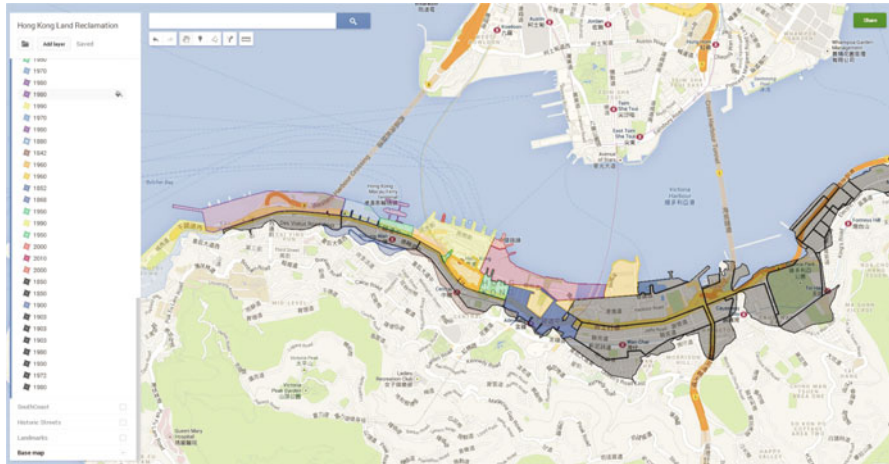


Fig. 2.5 Building an interactive visualization of land reclamation in Hong Kong from historical maps



Fig. 2.6 Using a variety of historical maps to recreate land reclamation records in Hong Kong

In a joint effort, the USC Shoah Foundation Visual History Archive (VHA) and the USC Spatial Sciences Institute built a demonstration tool to link historical maps to places mentioned in genocide survivors’ video testimonies in the VHA archive [Chi17a]. These linkages enrich the personal stories of the survivors by using the spatial and temporal context in historical maps to enable the viewers to “go back in time” to recreate the physical world of the historical experience of the

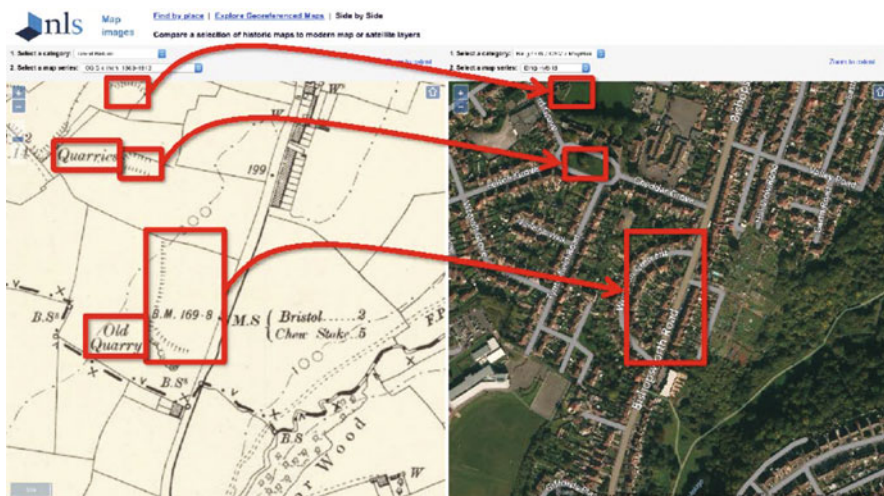


Fig. 2.7 Quarries and infill lands in a historical map and the contemporary satellite imagery



Fig. 2.8 Using historical maps to identify the wedding location (historical synagogue) described by the Holocaust survivor, Murray Burger

survivors (Fig. 2.8).²⁵ This work requires manually searching and going through large numbers and varieties of historical maps to find the place names mentioned in the interviews.

Nanetti et al. [Nan+15] manually transcribed and georeferenced the textual content in the Mappa Mundi by Fra Mauro (circa 1450). They used the transcribed

²⁵Peter Feigl’s Journey Through Historical Maps: <http://www.arcgis.com/apps/MapJournal/index.html?appid=6c3b4136b9304df09c9adcf86dd30dd5>.

data as a knowledge aggregator to represent the world as seen from Venice in the fifteenth centuries. They also plan to use the map data for automatic provenance and validation assessment of large and heterogeneous collections of other historical sources (e.g., books and diaries).

The example studies here demonstrate the important roles of historical maps in natural and social science as well as interdisciplinary research. They also show that not only extracting and recognizing map content is important, but providing semantic annotations to the map content and linking the map content to other data sources will enable researchers to investigate complex scientific problems (e.g., [Nan+15]) at a scale that cannot be done today.

2.3 Case Studies of Map Processing Technologies

Before the historical maps can be used in scientific investigations, an important process is map digitization involving a series of manual or computer assisted steps to convert map content into a machine-readable format. One of the most common types of map features for digitization is graphic symbols, which depict important and interesting geographic phenomena, such as wetlands (Fig. 2.9). The descriptive metadata of these symbols can be found in map labels or keys; however, labels are only capable of displaying limited information (e.g., place names) and keys provide categorical information. For example, Fig. 2.10 shows a group of unique buildings in a USGS topographic map, but the map does not provide any information about these buildings (e.g., names). Figure 2.11 shows a scanned map²⁶ of Baghdad, Iraq where most symbols are labeled with place names but retrieving and integrating further information (e.g., addresses) of these places from other sources requires additional efforts such as using the place names and locations to search on Wikipedia or DBpedia (a structured version of Wikipedia).²⁷

This section presents a semi-automatic and an automatic approach to solve the same problem: symbol spotting (or recognition) from historical maps. Both approaches use the SURF (Speeded Up Robust Features) matching framework [BTV06, Low+99] to demonstrate the end-to-end process. The SURF matching framework is supported by many computer vision libraries (e.g., OpenCV) and commonly used to locate and recognize objects in images with little training required (in contrast to deep learning approaches).

The first approach (Sect. 2.3.1) is a semi-automatic, training-by-example approach that requires minimal user effort and can handle various types of maps and symbols. One of the limitations of this approach is its capability to handle a large number of maps since each map would require some user intervention to generate

²⁶The map was purchased online from Gecko Maps and scanned to produce a map image with a 600 DPI (dots-per-inch) resolution.

²⁷<https://wiki.dbpedia.org>.

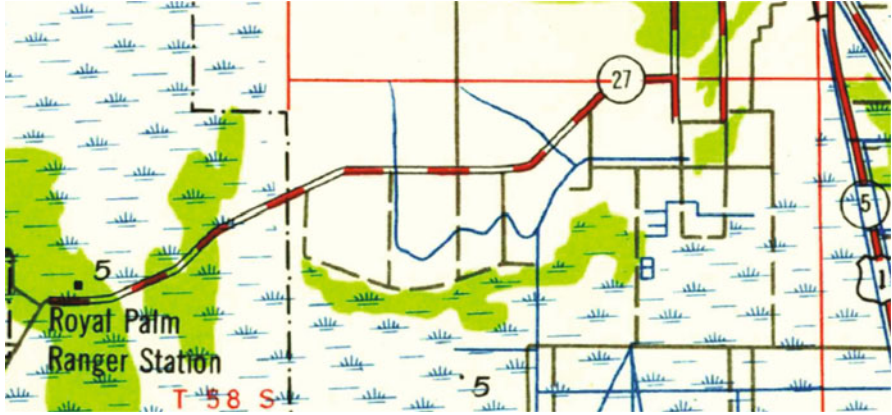
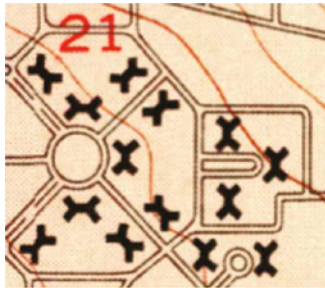


Fig. 2.9 Wetlands in a historical USGS topographic map (Miami, Florida, circa 1958)



(a)



(b)

Fig. 2.10 Buildings of the Park La Brea Apartment in a historical USGS topographic map (Hollywood, California, circa 1953) (a) and Google Earth imagery (b). (a) Buildings without text labels. (b) The Park La Brea apartment on Google earth imagery



Fig. 2.11 Symbols labeled with place names in a scanned Baghdad map (source: Gecko Maps)

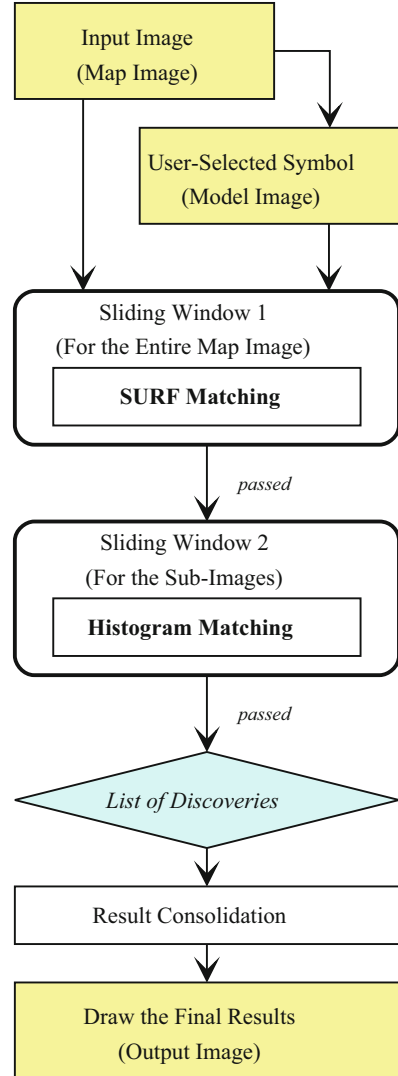
the training data. The second approach (Sect. 2.3.2) examines that when additional datasets are available, how the concept of geographic context can be used for the development of an effective way to fully automate the extraction of geographic information from scanned maps.

2.3.1 Case Study I: Semi-Automatic Symbol Recognition from Map Scans

This case study presents a semi-automatic, training-by-example approach, called SymbolRecognizer (Fig. 2.12), for spotting graphic symbols in map images. In the context of SymbolRecognizer, a model image is an image that covers a user-selected example in the input map (the red rectangle in Fig. 2.13). The recognition task is to search the map for symbols that match the model (i.e., target symbols). SymbolRecognizer utilizes a two-phase process: (1) using the SURF (Speeded Up Robust Features) matching [BTV06, Low+99] to efficiently identify the local regions (sub-images) where a target symbol might be present and (2) exploiting the pixel intensity distribution (with histogram matching) to verify the presence of a target symbol in each sub-image. Finally, SymbolRecognizer consolidates the identified symbols from the map and generates the final results.

Traditional document analysis techniques for spotting map symbols generally require a large number of training data, the presence of map keys (e.g., [SS98]), or

Fig. 2.12 The SymbolRecognizer framework



ad hoc preprocessing steps (e.g., image thresholding) [CLK14, Lla+01]. In contrast, SymbolRecognizer takes only one user-selected symbol example to extract the locations of all symbols that have a similar graphical appearance to the example. In addition, with the extracted symbol locations, this case study also demonstrates that SymbolRecognizer can efficiently enable automatic linkages between DBpedia records and map content.

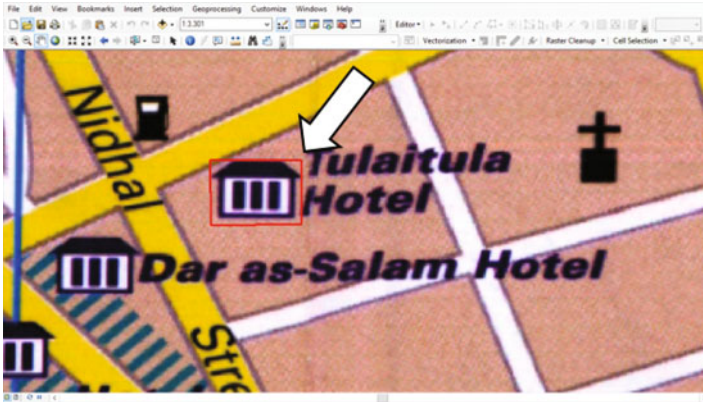


Fig. 2.13 A user-selected symbol example

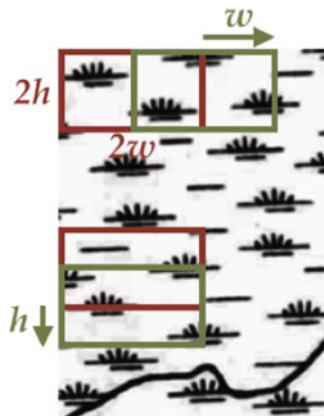


Fig. 2.14 The SURF matching sliding window

2.3.1.1 SURF (Speeded Up Robust Features) Matching

Considering a model image with a width and height equal to w and h pixels, in the first phase, SymbolRecognizer uses a sliding window of the size equal to $2w$ and $2h$ pixels and moves w or h pixels in the horizontal or vertical direction to scan through the entire input map (Fig. 2.14). The size of the sliding window guarantees that every target symbol is covered completely in at least one window (a sub-image). At each position of the sliding window, SymbolRecognizer detects the SURF descriptors from the sub-image and compares the detected descriptors with the SURF descriptors detected from the model image. The SURF descriptors are a type of scale-invariant feature descriptors that capture local “interest” points and their properties for image registration and object recognition. These interest points

describe the image intensity at specific pixels and the intensity differences between their adjacent pixels. If the comparison result contains a high number of matched features, the sub-image is highly likely to contain a target symbol²⁸ and is passed to the next phase.

2.3.1.2 Histogram Matching

The SURF matching is efficient and widely used to recognize real-world objects in photography or videos, but map symbols have simpler shapes (than real-world objects) and are relatively small, which can cause many false positives in the matching results. Therefore, to minimize the required training data (user-provided model images), SymbolRecognizer compares the pixel intensity²⁹ distributions of the model image and each sub-image that passes the SURF matching to determine whether or not a target symbol is present and to extract the symbol location.

For each sub-image that passes the first phase, SymbolRecognizer uses the model image to scan from the top-left corner and moves one pixel in the horizontal or vertical directions. Each scanning position records a similarity score calculated using the correlation of the grayscale histogram of the model image (H^{model}) and the grayscale histogram of the overlapping image patch (the overlapping area between the model image and the sub-image) (H^{patch}). The correlation is defined as follows:

$$\text{Similarity Score} = \frac{\sum_{i=0}^{255} (H_i^{model} - \overline{H^{model}})(H_i^{patch} - \overline{H^{patch}})}{\sqrt{\sum_{i=0}^{255} (H_i^{model} - \overline{H^{model}})^2 (H_i^{patch} - \overline{H^{patch}})^2}}. \quad (2.1)$$

SymbolRecognizer uses an empirically set threshold of 90% on the similarity score to filter out the sub-images that do not contain a target symbol and to locate the symbol location. If none of the scanning positions in a sub-image has a similarity score higher than 90%, SymbolRecognizer discards the sub-image; otherwise, the scanning position that has the highest similarity score (in a sub-image) is the detected location of a target symbol.

This method works on both color and grayscale images. SymbolRecognizer uses the grayscale histogram (0–255 luminosity levels) since the luminosity component is generally the most distinguishable component for separating geographic features in a map (i.e., maps need to be readable when printed with black and white printers).

²⁸The reader is referred to [Low+99] for details of this object recognition procedure.

²⁹For example, a white pixel will have a luminosity value of 0, and a black pixel will have an intensity value of 255.

2.3.1.3 Result Consolidation

SymbolRecognizer can detect a target symbol in overlapping sub-images during the SURF matching process since multiple sliding windows can cover a symbol more than once (Fig. 2.15a). To consolidate the results, if the detected target symbols overlap, SymbolRecognizer keeps only the symbol with the highest histogram matching score (Fig. 2.15b).

2.3.1.4 Results and Discussion

SymbolRecognizer was implemented as an Esri ArcMap plugin and was tested with maps from four sources. Table 2.1 shows the experiment results. Figures 2.11, 2.16, 2.17, and 2.18 show the test maps, and Figs. 2.16, 2.17, and 2.18 show sample results. For each test map, the user selected one sample symbol and SymbolRecognizer automatically processed the sample to find other symbols on the map.

Fig. 2.15 Result consolidation for overlapping sub-images. (a) Detected target symbols before consolidation. (b) Detected target symbols after consolidation

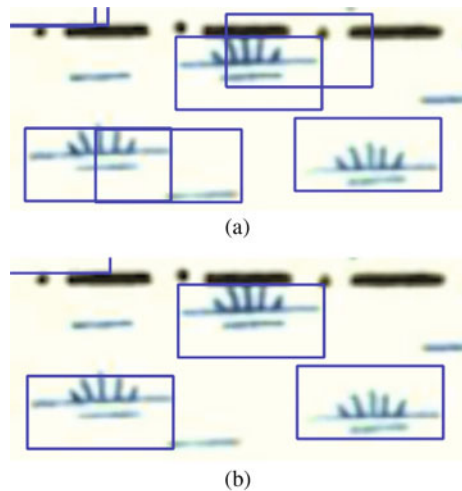


Table 2.1 SymbolRecognizer recognition results

Source	Image size (pixels)	# of target symbols	Precision	Recall
USGS Miami (1958)	409 × 438	87	97.33%	83.91%
USGS mine and mineral	2465 × 2150	25	100%	48%
USGS Hollywood (1953)	554 × 396	18	88.89%	88.89%
Gecko Maps, Baghdad	5104 × 2616	17	100%	88.23%

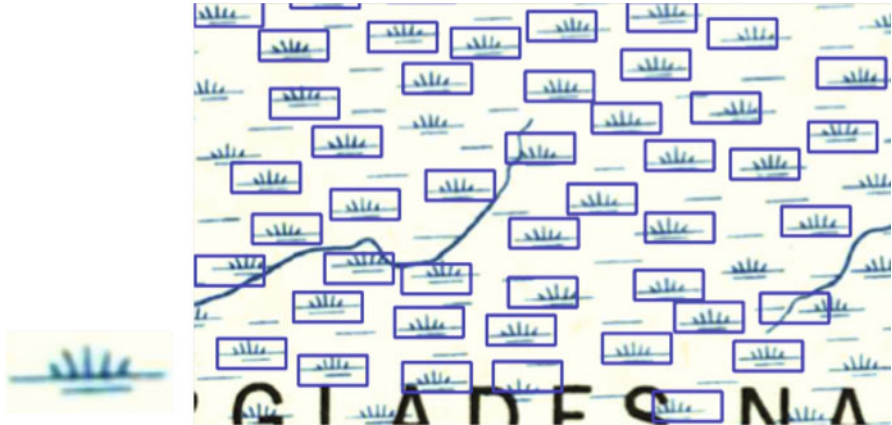


Fig. 2.16 A historical USGS topographic map (Miami, Florida, circa 1958); model images (left) and sample results where blue rectangles are the recognized locations (right)

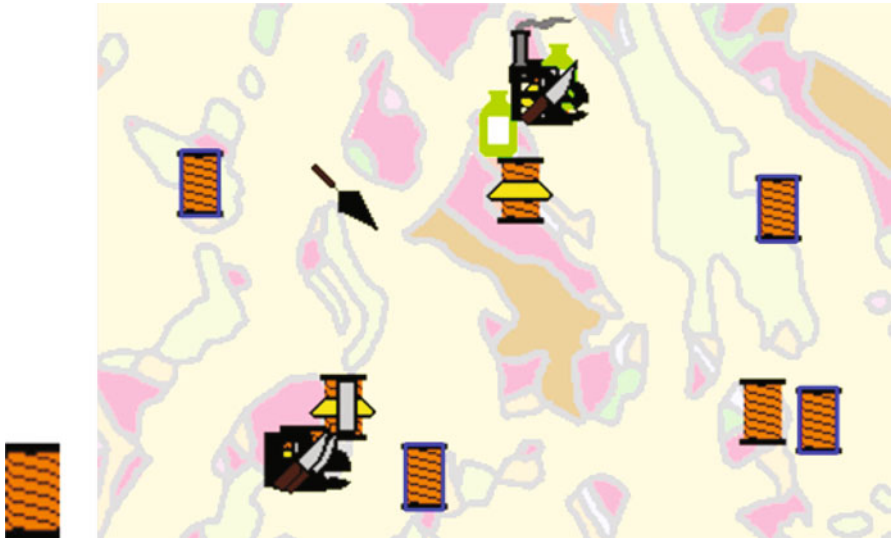


Fig. 2.17 The USGS mine and mineral processing plant locations map; model images (left) and sample results where blue rectangles are the recognized locations (right)

The results in Table 2.1³⁰ show promising extraction precision (with only a few false positives). The recognition results of the USGS Hollywood map have the

³⁰Precision is the number of correctly recognized symbols divided by the total number of recognized symbols. Recall is the number of correctly recognized symbols divided by the total number of symbols in the ground truth.

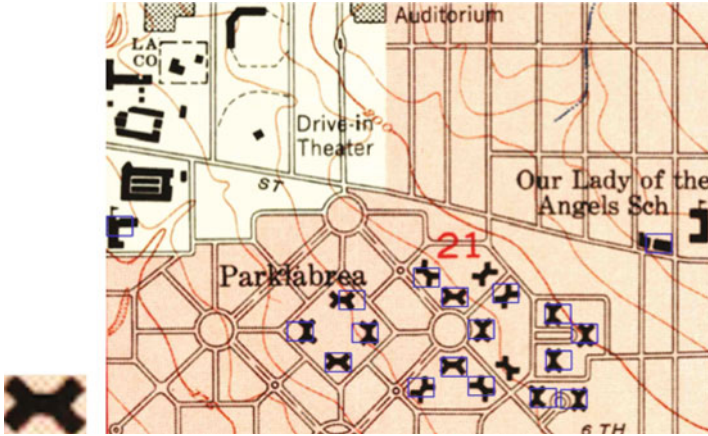


Fig. 2.18 A historical USGS topographic map (Hollywood, California, circa 1953); model images (left) and sample results where blue rectangles are the recognized locations (right)



Fig. 2.19 Examples of overlapping symbols

lowest precision number among all test maps since the target symbols (the Park La Brea apartment buildings) have varying orientations. Although the SURF matching is rotation invariant, the histogram matching process might generate true negatives if the image patch does not cover the entire symbol of different orientations in the sub-image. For all other test maps, the recognition precision are more than 97% since the symbols are in the same orientation.

For the recall of the recognition results, significantly overlapping features are the main cause of the true negatives. Figure 2.19 shows two examples of overlapping symbols in the USGS Mine and Mineral map. The overlapping symbol to the right was detected because only a small portion of the symbol was overlapped by another symbol. The symbol to the left was not detected since the entire symbol was almost covered by other symbols. The USGS Mine and Mineral map contains 12 (out of



Fig. 2.20 Automatically linking map symbols and DBpedia entries

25) significantly overlapping symbols and hence the extraction recall is the lowest among the test maps. For all other test maps, the recognition recall are greater than 83%.

The Baghdad map was manually georeferenced, and after the symbols were identified, SymbolRecognizer queried the DBpedia SPARQL endpoint³¹ to retrieve the nearby DBpedia entries to individual symbol geocoordinates. These entries can have various DBpedia types such as “Museum,” “Embassy,” “School,” and “Hotel.” Since the identified symbols were places in the same category, SymbolRecognizer links each symbol to a nearby DBpedia entry of the most popular DBpedia type among the retrieved entries. In this test area, the most popular category near the extracted symbol locations was “Hotel,” and there were only four hotel entries on DBpedia for the test area. Figure 2.20 shows the Baghdad map with the identified symbols linked with DBpedia URIs (Uniform Resource Identifier).

³¹<http://dbpedia.org/sparql>.

This case study shows that with a few user training samples (one for each test map), SymbolRecognizer can extract map symbols from a wide variety of maps. The experiment also demonstrates that once some of the map content is machine-readable, linking the map content with other data sources can be achieved by matching the spatial patterns of the machine-readable content (e.g., hotel locations) to other sources. This type of semi-automatic approach is ideal when the required number of maps for processing is small, and the variety of their content is significant.

2.3.2 Case Study II: Multi-Model, Context-Based Automatic Symbol Recognition from Map Scans

This case study presents an example of a fully automated recognition system, ContextSymbolRecognizer, for the recognition of hotel symbols from a scanned map using contextual information from a gazetteer. Given a gazetteer and the same Baghdad scanned map tested in Sect. 2.3.1, the task is to find all hotel locations in the map without any user intervention for training the underlying feature recognition algorithms, also assuming that the scanned map is georeferenced and the gazetteer does not contain all the hotels on the map.

2.3.2.1 Graphics Sampling Using Contextual Information

Figure 2.21 shows the overall process of ContextSymbolRecognizer. First, ContextSymbolRecognizer queries a gazetteer (e.g., GeoNames) using the map extent and a keyword to find relevant entries for the map symbols (building contextual information). In the case of the Baghdad map (Figs. 2.11), GeoNames contains two entries given the search keyword “hotel.” The two hotel locations are Baghdad Hotel (33.31867, 44.41516) and Palestine Hotel (33.31539, 44.41882).

Then ContextSymbolRecognizer uses the coordinates of these entries to crop the input map and generate training samples so that the user does not have to manually provide symbol examples. Specifically, ContextSymbolRecognizer locates these positions in the map and uses a buffer distance to crop the areas around these two point locations with the assumption that each of these two cropped areas contains at least one hotel symbol (Fig. 2.22) (adaptive graphics sampling). This step eliminates the need for user intervention. The crop distance is defined considering the map scale and feature type. Note that in Fig. 2.22, the second cropped sample contains more than one hotel symbol because the two hotels are close to each other.

Next, ContextSymbolRecognizer computes the SURF descriptors from the cropped areas and stores these descriptors as a knowledge base. Using the SURF descriptors derived from the samples, ContextSymbolRecognizer scans through the entire map to find map sub-sections that contain descriptors with similar values. If

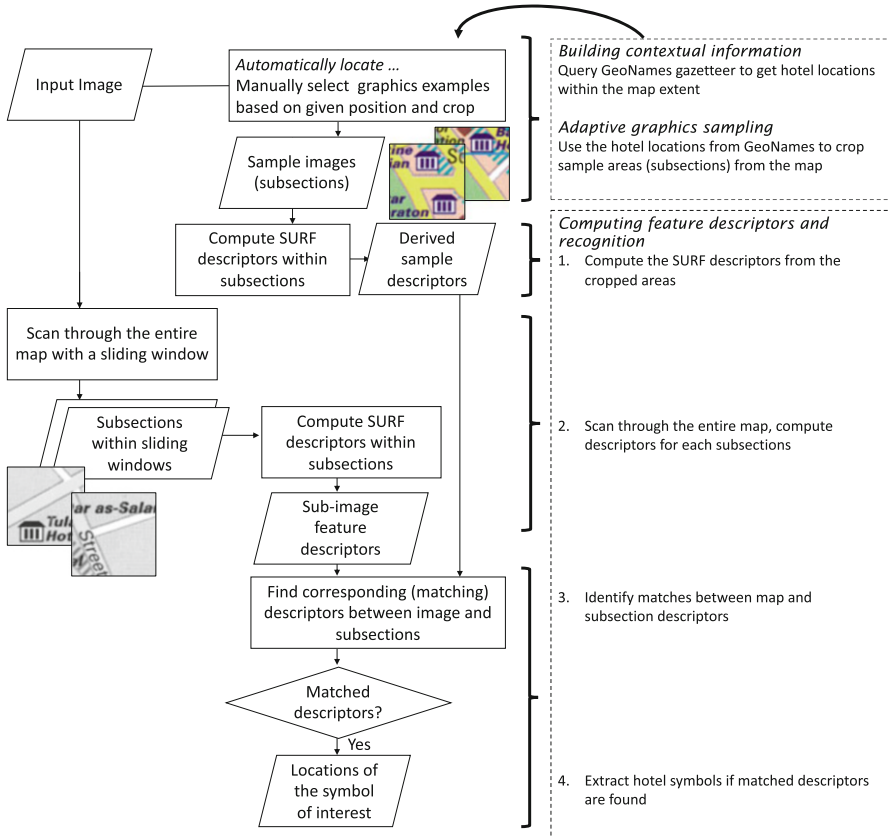


Fig. 2.21 The overall process for fully automated symbol spotting. (i.e., the process of “Manual select” graphic examples as in SymbolRecognizer is not required.) The process flow on the left shows a traditional approach for feature recognition using the SURF framework (e.g., SymbolRecognizer). ContextSymbolRecognizer eliminates the need for user-selected samples (model images) and thus fully automates the recognition process

certain combinations of descriptors are matched, ContextSymbolRecognizer labels these locations as hotel instances. This matching process using SURF for extracting map symbols is the same as in the semi-automatic SymbolRecognizer described in Sect. 2.3.1, but SymbolRecognizer relies on manually selected samples. In addition, ContextSymbolRecognizer does not require the historical matching step as in SymbolRecognizer since the external data source usually (e.g., hotel locations in GeoNames) provides more training data for extracting the SURF descriptors than the one manually provided sample as in the case study of SymbolRecognizer.

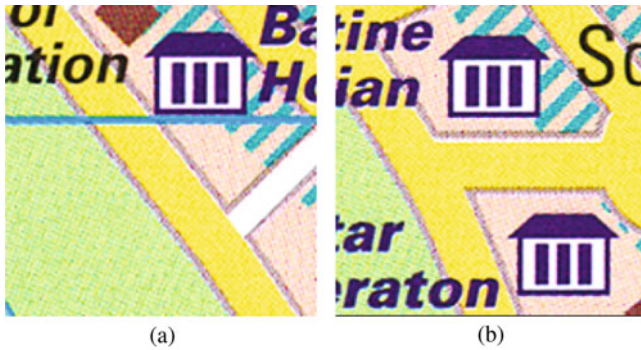


Fig. 2.22 Automatically cropped areas (center of these areas are the locations given in the gazetteer) that contain hotel symbols used as graphics examples to compute descriptors for extracting all other hotel symbols on the map: Baghdad Hotel (a) and Palestine Hotel (b)

2.3.2.2 Results and Discussion

This experiment was conducted on the scanned map covering Baghdad, Iraq in Fig. 2.11 as in the case study of SymbolRecognizer. ContextSymbolRecognizer recognized 13 hotel locations from the scanned map based on the two hotel entries found in the GeoNames gazetteer. Out of the 13 extracted hotels, 12 of them were correct (precision 92.3%). The total number of hotels in the map is 17 (i.e., the ground truth), thus the approach missed 5 hotel symbols (recall 70.58%). Figure 2.23 shows some examples of the extracted symbols. This result shows slightly lower precision and recall than that in the semi-automatic approach, SymbolRecognizer (Sect. 2.3.1), which was tested on the same task and map but involved user intervention to manually label one hotel sample in the map (100 and 88.23%, precision and recall, respectively).

The fact that ContextSymbolRecognizer was able to fully automatically recognize cartographic symbol using contextual information (here from GeoNames but also other data sources can be used, e.g., DBpedia) represents an important step forward in developing more capable recognition systems that scale well for massive data archives. This type of context-based approach uses existing knowledge of an area (describing some kind of geographic context) to guide the feature sampling and extraction process to eliminate user intervention. The ability to process maps without user intervention is necessary to exploit a large number of existing maps as well as the full richness of large-volume digital historical map archives. One limitation for ContextSymbolRecognizer is that the input maps have to be georeferenced, which makes it more suitable for digital map archives of georeferenced map series, such as the USGS archive of historical topographic maps.



Fig. 2.23 Sample extraction results from the fully automated hotel recognition case study. The small image on the right is one of the automatically identified hotel samples. The small white circles on the maps are the locations of the SURF descriptors. The yellow lines connect matches between the SURF descriptors of the map area and the sample. The samples in (a) show correct identified hotel symbols in four map areas. The sample in (b) shows the only incorrectly identified area

2.3.3 Case Study Discussion and Outlook

This section presented two case studies for symbol spotting (recognition) from map images. As described in Sect. 2.2, some studies only require a small number of maps, but these maps can come from a variety of sources and have a variety of graphical conditions (e.g., [NSC15]). In this case, the semi-automatic approach

is preferred since it can handle various types of maps and symbols. In contrast, for the studies that require a large number of maps from the same map series (e.g., generating historical place names from a complete historical map series for georeferencing museum collections), the fully automatic approach can be used since it has the potential to eliminate manual processing, labeling, and curating steps while achieving acceptable accuracy.

As the results indicate, the approach of context-based recognition has also the potential for processing large-volume map archives but would require a more complex framework and image recognition technologies (than SURF) (e.g., deep learning technologies) that enable the employment of geographic context and semantic modeling. One possible extension of these ideas is a recognition framework that exploits the fact that map content is not independent between editions in a map series (e.g., the USGS topographic map series) and change gradually and often cumulatively. One can build generic contextual information (e.g., hydrographic feature types, locations, and geometry) using existing contemporary map data (e.g., the USGS National Map layers) and use these contextual data to guide the feature extraction process in the most recent edition of a map similar to what is described in Sect. 2.3.2. The extracted data from the most recent edition can then be used as contextual information for the next (older) map edition in order to carry out the same recognition process. This way, older maps that suffer from low graphical quality can be processed using contextual information from map sheets that are close in time, and the results are expected to be more robust than with common recognition methods. Recent work [Dua+17, Uhl+17, Uhl+18b, Uhl+18a] has shown promising results of this context-based recognition approach using deep learning on the recognition of railroads, waterlines, and buildings from historical USGS topographic maps.

2.4 Chapter Summary

This chapter presented applications in natural and social sciences demonstrating the opportunities for the computer science and geographic information science communities to transform conventional research practices in using historical maps. For example, a new technology that automatically generates machine-readable or -understandable (e.g., LinkedData [BHB09]) place name databases from historical maps and to do so at scale³² will enable researchers in biology to minimize the time and effort for geolocating their data records and to efficiently query and analyze historical records by location and time. These opportunities also present unique possibilities for researchers in computer science and geographic information science to identify collaborators in broad scientific domains. This type of interdisciplinary

³²A computational process that can process a large number of maps within a reasonable amount of time.

collaboration allows researchers in computer science and geographic information science to create algorithms and applications to help solve “wicked” research problems and addressing real-world challenges facing our society. Further, the chapter discussed two case studies and their challenges in feature recognition from historical maps. The technologies in the case studies have already shown promising results in symbol recognition from heterogeneous historical maps though only using a previous generation computer vision technology (i.e., SURF descriptors). Further developments of these types of technologies will help make it possible to use a large number of heterogeneous historical maps efficiently and study historical geographic datasets on a large scale (e.g., [Dua+17, Uhl+17, Uhl+18b, Uhl+18a]).

References

- [Ada28] O.G. Adams, Place names in the north central counties of Missouri. Ph.D. Thesis, University of Missouri-Columbia (1928)
- [Ale+15] B. Alex, K. Byrne, C. Grover, R. Tobin, Adapting the Edinburgh Geoparser for historical georeferencing. *Int. J. Humanit. Arts Comput.* **9**(1), 15–35 (2015). <https://doi.org/10.3366/ijhac.2015.0136>
- [BHB09] C. Bizer, T. Heath, T. Berners-Lee, Linked data - the story so far. *Int. J. Semantic Web Inf. Syst.* **5**(3), 1–22 (2009). ISSN: 1552-6283
- [BTV06] H. Bay, T. Tuytelaars, L. Van Gool, SURF: speeded up robust features, in *European Conference on Computer Vision* (Springer, Berlin, 2006), pp. 404–417
- [Chi+16] Y.-Y. Chiang, S. Leyk, N.H. Nazari, S. Moghaddam, T.X. Tan, Assessing the impact of graphical quality on automatic text recognition in digital maps. *Comput. Geosci.* **93**, 21–35 (2016). <https://doi.org/10.1016/j.cageo.2016.04.013>. ISSN: 0098-3004
- [Chi17a] Y.-Y. Chiang, Linking historical maps to USC Shoah foundation visual history archive, in *the 28th International Cartographic Conference* (2017)
- [Chi17b] Y.-Y. Chiang, Unlocking textual content from historical maps - potentials and applications, trends, and outlooks, in *Recent Trends in Image Processing and Pattern Recognition*, ed. by K. Santosh, M. Hangarge, V. Bevilacqua, A. Negi (Springer, Singapore, 2017), pp. 111–124. ISBN: 978-981-10-4859-3
- [CLK14] Y.-Y. Chiang, S. Leyk, C.A. Knoblock, A survey of digital map processing techniques. *ACM Comput. Surv.* **47**(1), 1–44 (2014). <https://doi.org/10.1145/2557423>. ISSN: 0360-0300
- [Dav+15] C.C. Davis, C.G. Willis, B. Connolly, C. Kelly, A.M. Ellison, Herbarium records are reliable sources of phenological change driven by climate and provide novel insights into species’ phenological cueing mechanisms. *Am. J. Bot.* **102**(10), 1599–1609 (2015). <https://doi.org/10.3732/ajb.1500237>. ISSN: 0002-9122, 1537-2197
- [DBZ] C. D’Ignazio, R. Bhargava, E. Zuckerman, Cliff-clavin: determining geographic focus for news, in *NewsKDD: Data Science for News Publishing* (2014)
- [Dua+17] W. Duan, Y.-Y. Chiang, C.A. Knoblock, V. Jain, D. Feldman, J.H. Uhl, S. Leyk, Automatic alignment of geographic features in contemporary vector data and historical maps, in *Proceedings of the 1st Workshop on Artificial Intelligence and Deep Learning for Geographic Knowledge Discovery* (ACM, New York, 2017), pp. 45–54
- [GE07] I.N. Gregory, P.S. Ell, *Historical GIS: Technologies, Methodologies, and Scholarship*. Cambridge Studies in Historical Geography, vol. 39 (Cambridge University Press, Cambridge, 2007). ISBN: 9781139467711

- [GE15] B. Godfrey, H. Eveleth, An adaptable approach for generating vector features from scanned historical thematic maps using image enhancement and remote sensing techniques in a geographic information system. *J. Map Geograph. Libr.* **11**, 18–36 (2015). <https://doi.org/10.1080/15420353.2014.1001107>. ISSN: 1542-0353
- [Gre+15] I. Gregory, C. Donaldson, P. Murrieta-Flores, P. Rayson, Geoparsing, GIS, and textual analysis: current developments in spatial humanities research. *Int. J. Humanit. Arts Comput.* **9**(1), 1–14 (2015). <https://doi.org/10.3366/ijhac.2015.0135>
- [Gur+06] R.P. Guralnick, J.Wieczorek, R. Beaman, R.J. Hijmans, B.W. Group, et al., BioGeomancer: automated georeferencing to map the world's biodiversity data. *PLoS Biol.* **4**(11), e381 (2006). <https://doi.org/10.1371/journal.pbio.0040381>. ISSN: 1544-9173, 1545-7885
- [Hil+09] A.W. Hill, R. Guralnick, P. Flemons, R. Beaman, J. Wieczorek, A. Ranipeta, V. Chavan, D. Reimsen, Location, location, location: utilizing pipelines and services to more effectively georeference the world's biodiversity data. *BMC Bioinform.* **10**(suppl 14), S3 (2009). <https://doi.org/10.1186/1471-2105-10-S14-S3>. ISSN: 1471-2105
- [Kur13] L. Kurashige, Rethinking anti-immigrant racism: lessons from the Los Angeles vote on the 1920 Alien land law. *Southern California Quart.* **95**(3), 265–283 (2013). <https://doi.org/10.1525/scq.2013.95.3.265>. ISSN: 0038-3929
- [Lav13] C. Lavoie, Biological collections in an ever changing world: herbaria as tools for biogeographical and environmental studies. *Perspect. Plant Ecol. Evol. Systemat.* **15**(1), 68–76 (2013). <https://doi.org/10.1016/j.ppees.2012.10.002>. ISSN: 1433-8319
- [LL11] J.L. Leidner, M.D. Lieberman, Detecting geographical references in the form of place names and associated spatial natural language. *SIGSPATIAL Spec.* **3**(2), 5–11 (2011). <https://doi.org/10.1145/2047296.2047298>. ISSN: 1946-7729
- [Lla+01] J. Lladós, E. Valveny, G. Sánchez, E. Martí, Symbol recognition: current advances and perspectives, in *IAPR International Workshop on Graphics REcognition* (Springer, Berlin, 2001), pp. 104–127. ISBN: 9783540440666. https://doi.org/10.1007/3-540-45868-9_9
- [Low+99] D.G. Lowe et al., Object recognition from local scale-invariant features, in *Proceedings of the International Conference on Computer Vision, ICCV*, vol. 99 (1999), pp. 1150–1157
- [Man+17] S. Manson, J. Schroeder, D. Van Riper, S. Ruggles, et al., *IPUMS National Historical Geographic Information System: Version 12.0 [Database]*. (University of Minnesota, Minneapolis, 2017)
- [Mur+04] P.C. Murphey, R.P. Guralnick, R. Glaubitz, D. Neufeld, J.A. Ryan, Georeferencing of museum collections: a review of problems and automated tools, and the methodology developed by the Mountain and Plains Spatio-Temporal Database-Informatics Initiative (Mapstedi). *Phyloinformatics* **1**(3), 1–29 (2004)
- [Nan+15] A. Nanetti, A. Cattaneo, S.A. Cheong, C.-Y. Lin, Maps as knowledge aggregators: from Renaissance Italy Fra Mauro to web search engines. *Cartograph. J.* **522**, 159–167 (2015). <https://doi.org/10.1080/00087041.2015.1119472>. ISSN: 0008-7041
- [New10] T. Newbold, Applications and limitations of museum data for conservation and ecology, with particular attention to species distribution models. *Prog. Phys. Geogr.* **34**(1), 3–22 (2010). <https://doi.org/10.1177/0309133309355630>. ISSN: 0309-1333
- [NSC15] V. Ngo, J. Swift, Y.-Y. Chiang, Visualizing land reclamation in Hong Kong: a web application, in *Proceedings of the 27th International Cartographic Conference* (2015). ISBN: 9788588783119
- [PE10] G.H. Pyke, P.R. Ehrlich, Biological collections and ecological/environmental research: a review, some observations and a look to the future. *Biol. Rev. Camb. Philos. Soc.* **85**(2), 247–266 (2010). <https://doi.org/10.1111/j.1469-185X.2009.00098.x>. ISSN: 1464-7931
- [RB10] N.E. Rios, H.L. Bart, *GEOlocate (Version 3.22) [Computer Software]*. Belle Chasse, LA, Tulane University Museum of Natural History (2010)

- [Sam+13] G. Samy, V. Chavan, A.H. Ariño, J. Otegui, D. Hobern, R. Sood, E. Robles, Content assessment of the primary biodiversity data published through GBIF network: status, challenges and potentials. *Biodivers. Inform.* **82** (2013). <https://doi.org/10.17161/bi.v8i2.4124>. ISSN: 1546-9735
- [SS98] H. Samet, A. Soffer, MAGELLAN: map acquisition of geographic labels by legend analysis. *Int. J. Doc. Anal. Recognit.* **1**(2), 89–101 (1998). <https://doi.org/10.1007/s100320050009>
- [Uhl+17] J.H. Uhl, S. Leyk, Y.-Y. Chiang, W. Duan, C.A. Knoblock, Extracting human settlement footprint from historical topographic map series using context-based machine learning, in *IET Conference Proceedings* (2017)
- [Uhl+18a] J.H. Uhl, S. Leyk, Y.-Y. Chiang, W. Duan, C.A. Knoblock, Spatialising uncertainty in image segmentation using weakly supervised convolutional neural networks: a case study from historical map processing. *IET Image Process.* **12**(11), 2084–2091 (2018)
- [Uhl+18b] J. Uhl, S. Leyk, Y.-Y. Chiang, W. Duan, C. Knoblock, Map archive mining: visual-analytical approaches to explore large historical map collections. *ISPRS Int. J. Geo Inf.* **7**(4), 148 (2018)
- [Vel+13] M. Vellend, C.D. Brown, H.M. Kharouba, J.L. McCune, I.H. Myers-Smith, Historical ecology: using unconventional data sources to test for effects of global environmental change. *Am. J. Bot.* **1007**, 1294–1305 (2013). <https://doi.org/10.3732/ajb.1200503>. ISSN: 0002-9122, 1537-2197
- [Yos+14] K. Yoshida, H.A. Burbano, J. Krause, M. Thines, D. Weigel, S. Kamoun, Mining herbaria for plant pathogen genomes: back to the future. *PLoS Pathog.* **10**(4), e1004028 (2014). <https://doi.org/10.1371/journal.ppat.1004028>. ISSN: 1553-7366, 1553-7374
- [Yu14] J. Yu, *Coastline of Hong Kong Island* (Chung Hua Publisher, Beijing, 2014). ISBN: 9888394762

Chapter 3

Creating Structured, Linked Geographic Data from Historical Maps: Challenges and Trends



Abstract Historical geographic data are essential for a variety of studies of cancer and environmental epidemiology, urbanization, and landscape ecology. However, existing data sources typically contain only contemporary information. Historical maps hold a great deal of detailed geographic information at various times in the past. Yet, finding relevant maps is difficult, and the map content is not machine-readable. This chapter presents the challenges and trends in building a map processing, modeling, linking, and publishing framework. The framework will enable querying historical map collections as a unified and structured spatiotemporal source in which individual geographic phenomena (extracted from maps) are modeled (described) with semantic descriptions and linked to other data sources (e.g., DBpedia). This framework will allow making use of historical geographic datasets from a variety of maps, efficiently, over large geographic extents. Realizing such a framework poses significant research challenges in multiple fields in computer science including digital map processing, data integration, and the Semantic Web technologies, and other disciplines such as spatial, social, and health sciences. Tackling these challenges will not only advance research in computer science and geographic information science but also present a unique opportunity for interdisciplinary research.

3.1 Introduction

Historical geographic datasets and historical Geographic Information Systems (GISs) support a variety of studies such in environmental epidemiology, urbanization, and landscape ecology (e.g., [GE07]), but existing data sources (e.g., gazetteers) typically contain only contemporary information. Historical maps are a great source of geographic information in the past (e.g., historical place names, landmarks, and transportation networks) and are often the only source that provides professionally surveyed historical data about the natural and built environment. Today, map archives such as the USGS (United States Geological Survey) National

Geologic Map Database,¹ USGS Topographic Maps,² David Rumsey Map Collection,³ OldMapsOnline.org,⁴ and the National Library of Scotland⁵ store a large amount of historical maps in either paper or scanned formats (e.g., as tiled Web maps or downloadable scanned images). However, only a small portion of these historical maps is georeferenced, and even fewer of them have searchable metadata or machine-readable content. This prevents historical maps from being indexed and searched and limits the opportunity for both researchers and the general public to access valuable historical information in the maps.

Even with the long-term effort and recent advances in map processing techniques (e.g., [Chi10, CLK14, Bud18, Uhl19]), making a large number of historical maps searchable (by location, time, keywords, and event topics) and their content useable in an analytic environment (e.g., in a GIS) is still prohibitively expensive and time-consuming. As a result, studies that require accurate historical information are usually limited to process only a few historical maps and examine a small area or a short time period for which manual data curation is possible. Beattie [Bea14] created a three-dimensional historical topography of the Ballona Creek watershed (Marina del Rey, California) from two historical USGS topographic maps (circa 1896 and 1902). This historical topography enables environmental planners to compare historical and current conditions of the Ballona Creek watershed to identify landscape changes. Kurashige [Kur13] used historical census data, voting records, and the precinct boundaries extracted from a 1920 map to study “who” (e.g., occupations and political parties) in Los Angeles voted for the 1920 California Alien Land Law that discriminates against the Japanese. In other cases, the studies that require geographic information in the past but do not have access to appropriate datasets would approximate historical information using contemporary datasets. For example, the Yellow-Star Houses project identified 1944 addresses of designated compulsory residences in Budapest from historical decrees (circa 1944), used a contemporary street dataset to geolocate these historical addresses, and mapped them on Google Maps. The contemporary street dataset provides approximate locations of these designated compulsory residences. While in this case, Google Maps serves a convenient visualization platform, using a historical map as the basemap can further contribute rich geographic information in the past, such as nearby transportation hubs at the time (Fig. 3.1). Further, like many other disciplines when data publication tools are not easy to use or not available, the manual curation results from these studies often do not attach to the research publications and only exist on the Web with simple descriptions or in data repositories that are difficult to find.

¹http://ngmdb.usgs.gov/ngmdb/ngmdb_home.html.

²<http://ngmdb.usgs.gov/maps/TopoView/>.

³<http://www.davidrumsey.com/>.

⁴<http://OldMapsOnline.org/>.

⁵<http://maps.nls.uk/>.



Fig. 3.1 Yellow-star houses on Google Maps (top); a section of a historical map of Budapest, Hungary (circa 1941) showing the transportation hub (middle and bottom) (sources: <http://www.yellowstarhouses.org/> (top) and <http://riowang.blogspot.com/2011/02/after-siege.html>) (bottom))

This chapter⁶ presents the challenges and trends for building a map processing, modeling, linking, and publishing framework that allows querying collections of large numbers and varieties of historical maps as a structured, linked geographic data source in which individual geographic phenomena (extracted from maps) are modeled with semantic descriptions and linked to other data sources (e.g., GeoNames⁷ and DBpedia⁸). The term “semantic description” here refers to the metadata composed with controlled vocabularies capturing the intra-relationships between geographic phenomena within a map (e.g., an infill land near a quarry could be a waste disposal site) and the inter-relationship between historical map data and the huge amount of LinkedData [BHB09] already published on the Internet. This framework makes it possible to efficiently study historical spatiotemporal phenomena on a large scale (both in time and space) and solve problems that cannot be easily answered with only contemporary geographic datasets.

Figure 3.2 shows an example system if the proposed framework is successful. Figure 3.2a shows a portion of a historical Ordnance Survey six-inch map (circa 1902) and contemporary satellite imagery of the same area (Bristol, UK). The historical map shows two quarry locations and infill lands (the red rectangles). Quarries are a common pollution source (at which the polluted materials could be dumped at nearby infill lands). This type of contamination could make the soil not suitable for growing edible vegetables and fruits. Figure 3.2b shows a current farming area on the potentially contaminated land (the red polygon). The question at hand is whether or not it is safe to grow grapefruit in this area. The modeled historical geographic data from the proposed framework will support the following reasoning process. Possible contamination materials from a 1902 quarry are heavy metals M . The areas of infill lands extracted from the historical map are modeled as a probability surface of M (i.e., the target region). The accumulated rain precipitation over the target region from 1902 to 2015 is R . The main soil type of the target region is S . The probability of that the target region still contains M given the probability surface, R , and the soil type, S , is low, so the target region can be used to grow edible plants. Since grapefruit trees have a shallow root system, growing a grapefruit tree may be safe in the target area.⁹

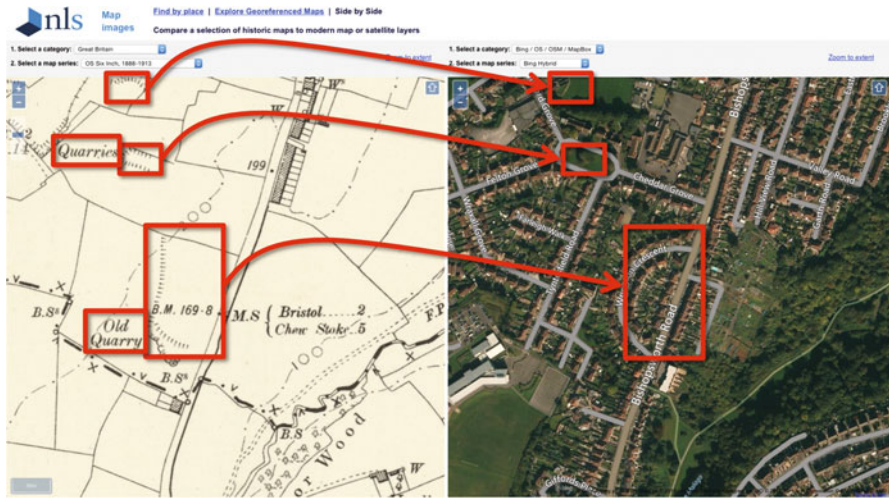
Realizing the proposed framework poses significant research challenges in multiple fields in computer science and geographic information science, including digital map processing, data integration, and Semantic Web technologies. Figure 3.3 shows an example implementation of the proposed framework. First, there are

⁶This chapter is based on a previous vision paper presented at the 2015 ACM SIGSPATIAL Conference [Chi15] and the First Place of the Best Vision Paper Award sponsored by the Computing Research Association’s Computing Community Consortium under the CCC Blue Sky initiative.

⁷<https://www.geonames.org/>.

⁸<http://wiki.dbpedia.org/>.

⁹This is just an example. By no means the author is an expert of soil contamination or growing grapefruits.



(a)



(b)

Fig. 3.2 Using historical maps to identify historical contamination sites (source: <http://maps.nls.uk/geo/explore/sidebyside.cfm#zoom=17&lat=51.4235&lon=-2.6157&layers=6&right=BingHyb>). (a) Locations of quarries and infill lands in a historical ordnance survey six-inch map (Somerset VI.SW, circa 1902; Somerset, South West England, UK) and the contemporary satellite imagery. (b) Current farming areas (approximated by the red polygon) that could have been affected by the historical contamination site

automatic map processing services that can process a large number of historical maps automatically to convert (some of) their content to a structured, machine-readable format. For the maps that are difficult to process with automatic services,

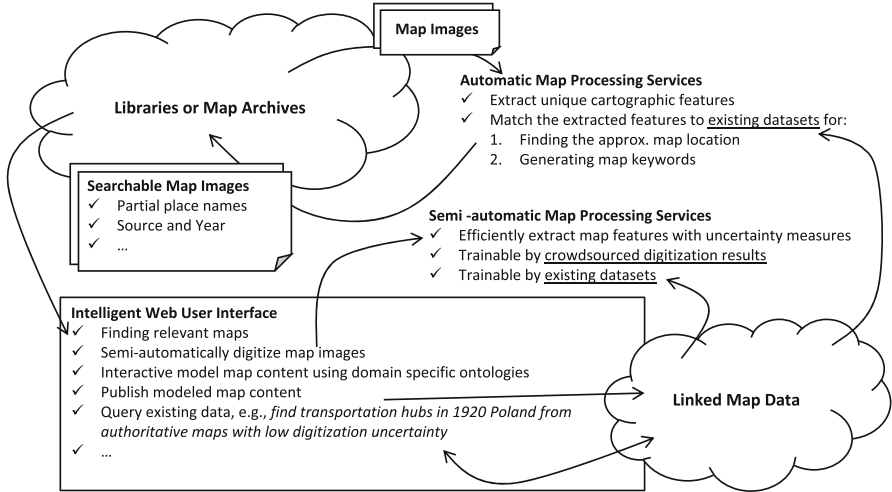


Fig. 3.3 An example implementation of the proposed map processing, modeling, linking, and publishing framework

there are *semi-automatic map processing services* supported by *intelligent Web user interfaces* to help facilitate manual digitization (e.g., crowdsourcing digitization projects, post-processing for improving the results from an automatic map processing system). The extracted structured data then can serve as metadata to support map search (i.e., *searchable map images*) and can be used to match to other sources for linking the map content to other data sources, adding additional metadata to the maps, and georeferencing the maps.

The challenges for realizing this framework include three interrelated challenges (not in a particular order):

- how to make historical maps easily searchable,
- how to efficiently and accurately convert map content to a machine-readable format and record provenance information and potential conversion errors, and
- how to generate semantic descriptions for historical map images, their content, and digitization related information (e.g., uncertainty), and link them to other data sources.

The following sections explain the challenges, research directions, and technology trends for overcoming these challenges. The remainder of this chapter is structured as follows: Section 3.2 describes the challenges and technologies supporting effective map searches. Section 3.3 presents an overview of various types of technologies for converting historical map scans to machine-readable, structured formats. Section 3.4 provides a brief background of geographic data formats and discusses relevant tools for modeling and linking geographic data. Section 3.5 summarizes the chapter and discusses future work.

3.2 Finding Relevant Historical Maps

Finding relevant historical maps given a region, a point in time, or a topic is never easy.¹⁰ This is because most of the historical maps in the libraries, map archives, and on the Web are usually scanned images and stored and maintained with limited metadata. The challenge here is how to generate some metadata automatically for large numbers and wide varieties of maps to support effective map searches. For example, the metadata could contain basic information, including the map geocoordinates and publication and revision dates, to enable spatiotemporal map queries, of a comprehensive collection of information, including the coordinate systems and projection information, data sources, the publisher, and place names. The maps can be any map images on the Web or can come from a specific map series in a map archive.

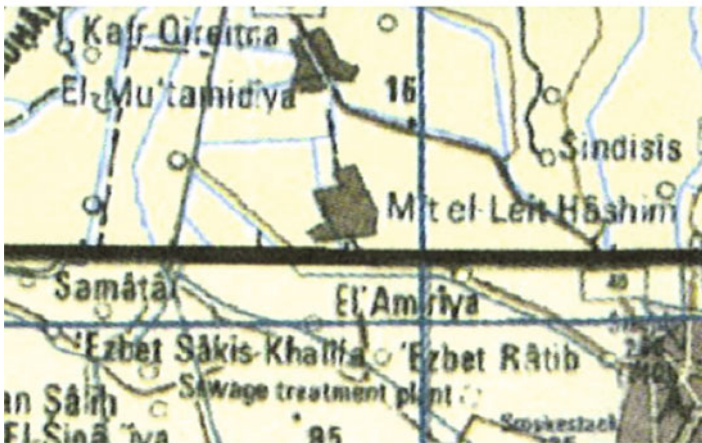
A straightforward way to enable effective map search is to apply optical character recognition (OCR) technologies on the map images to extract their text content (e.g., [Chi13, Wei13, CK14, Wei17, Chi17, LLZ18]). Compared with OCR in other types of document images, detecting and recognizing map text has to address particular challenges. This is because historical place name dictionaries (gazetteers) that can be used to correct partially recognized words are often unavailable. Also, map content is complex and contains overlapping feature layers. Further, the quality of the original paper maps and the scanning process (e.g., map fonts and scan resolution) can have a crucial impact on the OCR accuracy [Chi+16] (Fig. 3.4). In particular, Weinman [Wei17] developed a novel approach that exploits prior knowledge of the placement of text labels and geographic features as well as the typographic style consistency of the labels to improve the OCR results. Moreover, after OCR, the recognized text content becomes keywords describing each map, but the recognition results are typically separate words that require additional treatments to constitute complete place names and thus support meaningful search queries [LC18b]. For example, the individual words “Los” and “Angeles” by itself provide significantly less descriptive information compared to the complete place name “Los Angeles.”

Instead of directly recognizing text labels from historical maps to generate metadata, another approach is to automatically link the maps to other datasets and then use the linked datasets to generate metadata. For example, in [Wei13, Wei17], the author presents methods that recognize text labels in maps to then match the locations and text content of the recognized labels to a gazetteer of contemporary names for georeferencing the maps. While this approach shows promising results with small-scale maps (for which most of the map text exist in the gazetteer), many interesting maps are large-scale maps containing historical place names that are not in the gazetteer. In addition to text matching, one other direction is to extract distinctive shapes and spatial patterns of geographic features from

¹⁰To appreciate this difficulty from experience, the reader is encouraged to explore how long it would take to find a large-scale map of 1941 Budapest.



(a)



(b)

Fig. 3.4 Maps with common typewritten fonts usually show the best results for OCR compared to maps with less common fonts (e.g., Fraktur, Antiqua) or stenciled and handwritten text. Also, text in maps often overlaps with other map layers, which makes text recognition particularly sensitive to general printing quality issues, such as blurring, false coloring, and mixed colors. (Figure adapted from [Chi+16]). (a) Stenciled text in a historical map of Denmark. (b) An example of poor print quality in a National Imagery and Mapping Agency (NIMA) evasion chart (EVC NH-36A, NIMA ref. no. EVCXXNH36A)

maps automatically (e.g., road intersections [CK08] and contour lines [Pez11]) and then search for matching patterns in existing datasets to link map images to the datasets (e.g., [Che+04]). The challenge for this feature-matching approach is trifold. First, the automatic approaches for geographic feature extraction could be sensitive to the quality of the input map (similar as for text recognition) and hence

would not generate sufficiently distinctive feature patterns [Chi+15]. Second, the matching process could be computationally expensive if the search space is large. A combination of text and geographic feature matching could help to overcome these difficulties by providing different types of matching candidates and reducing the search space (e.g., only search for a match in the regions that contain at least 50% of the recognized text). Third, distinctive shapes of geographic features can be difficult to find and might not be consistent across multiple data sources due to the inconsistency in coordinate/projection systems and the varying degrees of map generalization. To overcome this difficulty of geometry inconsistency for feature matching, recent work employs a reinforcement learning strategy to automatically align contemporary vector data to historical maps [Dua+17]. The reinforcement learning framework enables informed search for corresponding features and hence makes the alignment process efficient. The framework does not require a preprocessing step for the extraction of distinctive shapes and patterns of geographic features (e.g., road intersections) (Fig. 3.5).

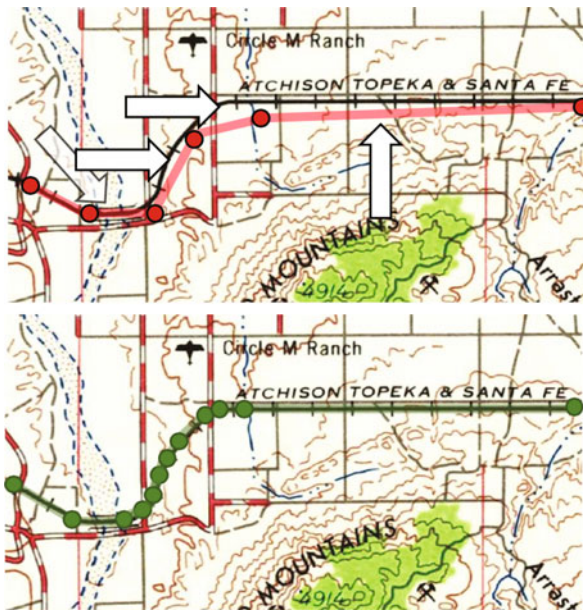


Fig. 3.5 The contemporary vector data of railroads (red vector lines, top image) could have misalignment (locations pointed by the white arrows, top image) and different feature representations than the features on the map. The automatic alignment system learns [Dua+17] how to approximate the geometry of map features and align the contemporary vector data to the map in a reinforcement learning environment that breaks the contemporary vector data into small segments and iteratively moves them towards the map features. This way the automatic alignment system does not require the extraction of distinctive shapes and patterns of geographic features

After a map is linked with additional datasets, the next challenge is how to generate the metadata to enable effective map searches. Gelernter [Gel08] built an automatic text mining approach to find maps in journal articles. This approach classifies the maps by years and themes using their companion text. This work shows that when enough data are linked to a map image, the data can be used to generate a comprehensive set of metadata to support meaningful search queries.

3.3 Converting Map Content to Machine-Readable Formats and Record Uncertainty

The mainstream approach for converting the image content of maps to a structured, machine-readable format still heavily relies on manual work with some help from raster-to-vector conversion software (e.g., R2V¹¹) or a GIS. This is because automatic map processing algorithms and systems are often not robust to process a wide variety of maps and typically requires advanced knowledge in computer science to train and use the system [CLK14]. For example, automatic systems for recognizing text features from maps often result in partially recognized strings because map labels often overlap with other map features, such as road lines and do not follow a fixed orientation within a map [Nag+97, CK14]. Moreover, many historical scanned maps suffer from poor graphical quality due to the bleaching of the original paper maps and archiving practices. However, manual map digitization requires a tremendous amount of time and knowledge in GIS. Beattie [Bea14] spent more than 70 h on manual tasks for extracting contour lines from two USGS historical maps (which also requires the knowledge of various tools in image processing and GIS). Godfrey and Eveleth [GE15] demonstrated a GIS workflow for digitizing a 1986 Idaho map for displaying the map information in a Web environment.

To scale up manual digitization efforts and to handle the vast variety of historical map types, crowdsourcing with semi-automatic approaches for map digitization has been shown to be more robust (than fully automatic approaches) in producing accurate results [CLK14]. The challenges here include how to automatically verify user inputs and how to build adaptive semi-automatic techniques that increase the level of automation as more maps are processed and eventually eliminate manual work once enough samples are processed. Both the British Library and the David Rumsey Map Collection held events to georeference their map collections by crowdsourcing. The New York Public Library's crowdsourcing approach for map digitization went one step further in that it provided semi-automatic tools for extracting parcel polygons from US insurance maps.¹² They also noted that a

¹¹<http://ablesw.com/r2v/download.html>.

¹²<https://github.com/NYPL/map-vectorizer/>.

fully manual approach would not scale to process their map collections within a reasonable time [Art13].

Also, while there exists an abundant work on automatic and semi-automatic map processing techniques [CLK14], only a few studies go beyond raster-to-vector conversion to record the processing “uncertainty” during the extraction. As noted in an earlier technical report from the Aeronautical Chart and Information Center [GS62], accuracy of the source material, intermediate and final products need to be considered to achieve the optimum utilization of a map product. To estimate the accuracy of the final datasets, the challenge is how to build systematic and objective evaluation methods for individual steps in a map processing tool and produce a final accuracy estimate [LBW05]. In a recent work, Lin and Chiang [LC18a] built an approach that systematically quantifies the uncertainty measurement resulting from an image recognition model (Convolutional Neural Networks) and the centerline extraction process (the thinning operator) for extracting the centerlines from area features in a historical map. The uncertainty measurement of the extracted centerlines enables automatic conflation between the extracted map data (in a vector format) and contemporary vector data. In contrast, vector-to-vector conflation often relies on a user-defined buffer distance to identify the matching features from the sources for conflation.

The following subsections provide an overview of the challenges and current trends in digital map processing, including crowdsourcing approaches (Sect. 3.3.2), semi-automatic approaches (Sect. 3.3.2), and fully automatic approaches with the help of contextual geographic data (Sect. 3.3.3).

3.3.1 Crowdsourcing Approaches

Crowdsourcing is not a new idea (but can be very difficult to implement and popularize) in document recognition and map processing. In addition to the crowdsourcing events held by the David Rumsey Map Collection and the New York Public Library, specifically, for converting textual content in historical maps using crowdsourcing, the Pelagios Commons¹³ is a notable community that provides tools and online infrastructure to facilitate annotating historical locality references in digital materials. Their tools allow semi-automatic extraction, recognition, annotation, and linking of place names in historical maps [Sim+10, Sim+14, S+15]. These tools and the online infrastructure allow them to provide full-text searchable place data ranges from ancient times to 1500 AD and from Europe to East Asia. Another example of crowdsourcing projects for historical map digitization is the GB1900 project.¹⁴ With crowdsourcing, the GB1900 project now provides about two million historical place names and other text (e.g., Fig. 3.6) from the six-inch

¹³<http://commons.pelagios.org/>.

¹⁴<http://www.visionofbritain.org.uk/data/#tabgb1900>.

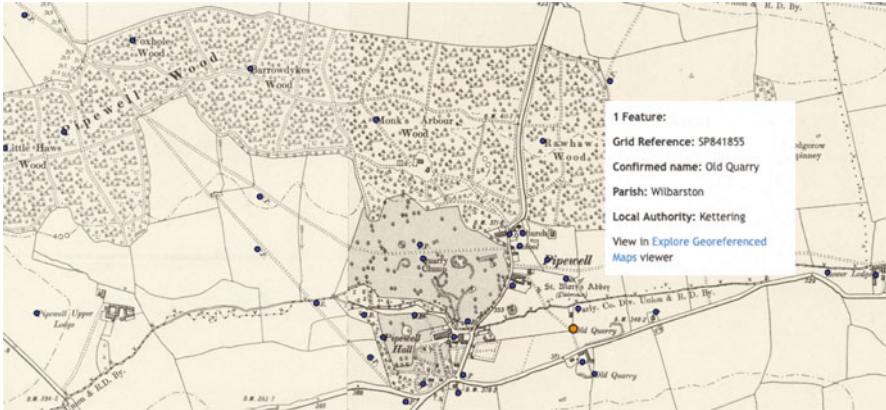


Fig. 3.6 An example of the GB1900 dataset where the blue dots show the locations of digitized text, and the orange dot is the selected text (source: <https://geo.nls.uk/maps/gb1900/#zoom=16&lat=52.4636&lon=-0.7627&layer=0>)

maps (Second Edition County Series) published by the Ordnance Survey between 1888 and 1914 covering the entire Great Britain.

When the crowdsourcing strategy is used, approaches for cross-validating between the user-generated content and the “gold data” as well as recording the provenance information are required for quality control (e.g., [GGH15, Bar+18, Bud+16]). As an example of quality control technologies, in the New York Public Library crowdsourcing project, the participants were asked to provide polygon outlines for building footprints from the nineteenth and early twentieth century historical maps using a Web interface. Many participants could provide polygon outlines for the same building footprint, and hence these crowdsourced polygon outlines are not guaranteed to be consistent. Budig et al. [Bud+16] developed an approach that takes multiple participant inputs of polygon outlines of the same building footprint from the New York Public Library crowdsourcing project and generates a “consensus polygon” that is similar to the majority of all user inputs. Their approach was tested on approximately 3000 polygon outlines for 200 building footprints and the “consensus polygon” was correct 96% of the time. In addition to using the user contributions (e.g., the polygon outlines) for quality control, Barz et al. [Bar+18] proposed a novel unsupervised approach to rank participants of a crowdsourcing project using a similar idea to PageRank [Pag+99]. This unsupervised approach computes a user reputation score for individual users based on the consistency of their contributions. The user reputation score helps in quality control as the contributions from users with a higher score is more trustworthy (accurate) than the users with a lower score.

3.3.2 *Semi-automatic Approaches*

To handle a wide variety of maps, the state-of-the-art graphics recognition technologies for cartographic symbol extraction from scanned maps rely on a user labeling process to generate a set of shape and/or color descriptors for detecting the feature of interest (e.g., [CLK13, CCM14, FK97, FC97, KZ03, Bud18]).

Chiang, Leyk, and Knoblock [CLK13] developed an interactive approach that uses manually collected road and non-road (e.g., wetland) samples to extract road vector data and remove noise from historical USGS maps (Fig. 3.7). This approach reduced 38% of the overall processing time with accurate results (compared to manual digitization), but it still required 50 min (including sample labeling and result curating) to process a map of 2283×2608 pixels.¹⁵ While this approach has demonstrated the ability to produce accurate results, the manual sample labeling would be required for processing each individual map sheet as paper maps are printed (archival) documents and often suffer from poor graphical conditions [KZ03]. The significance of these image quality issues can vary from one map area to another, requiring additional user labeling, and the already collected samples may not be directly applicable to another map.

In a series of publications [BDW16, Bud18, BvK16, Bud16, BD15], the authors developed a number of applications using active learning technologies for processing historical maps and early prints, such as extracting map text [Bud16, BD15] and linking text and map symbols [BDW16]. These applications include efficient and intelligent user interaction strategies to interactively learn to improve the system for the document processing tasks.

Bastani et al. [Bas+18] present a “machine-assisted map editing” system, MAiD, that integrates intelligent capabilities to an OpenStreetMap online editor. MAiD helps to reduce human efforts in tracing road centerlines for imagery digitization by automatically inferring road segments and only asking for human validation. The system, while designed for processing imagery data, could be used as a semi-automatic data extraction tool for historical maps to bridge the gap between automatic feature recognition systems and manual curation work.

This line of semi-automatic approaches has shown promising results on single map sheets. However, because they require user inputs, these approaches do not scale well for processing large numbers of various types of historical maps with heterogeneous contents.

¹⁵A USGS historical topographic map with the 600 DPI (dots-per-inch) scan resolution is about $12,000 \times 12,000$ pixels.

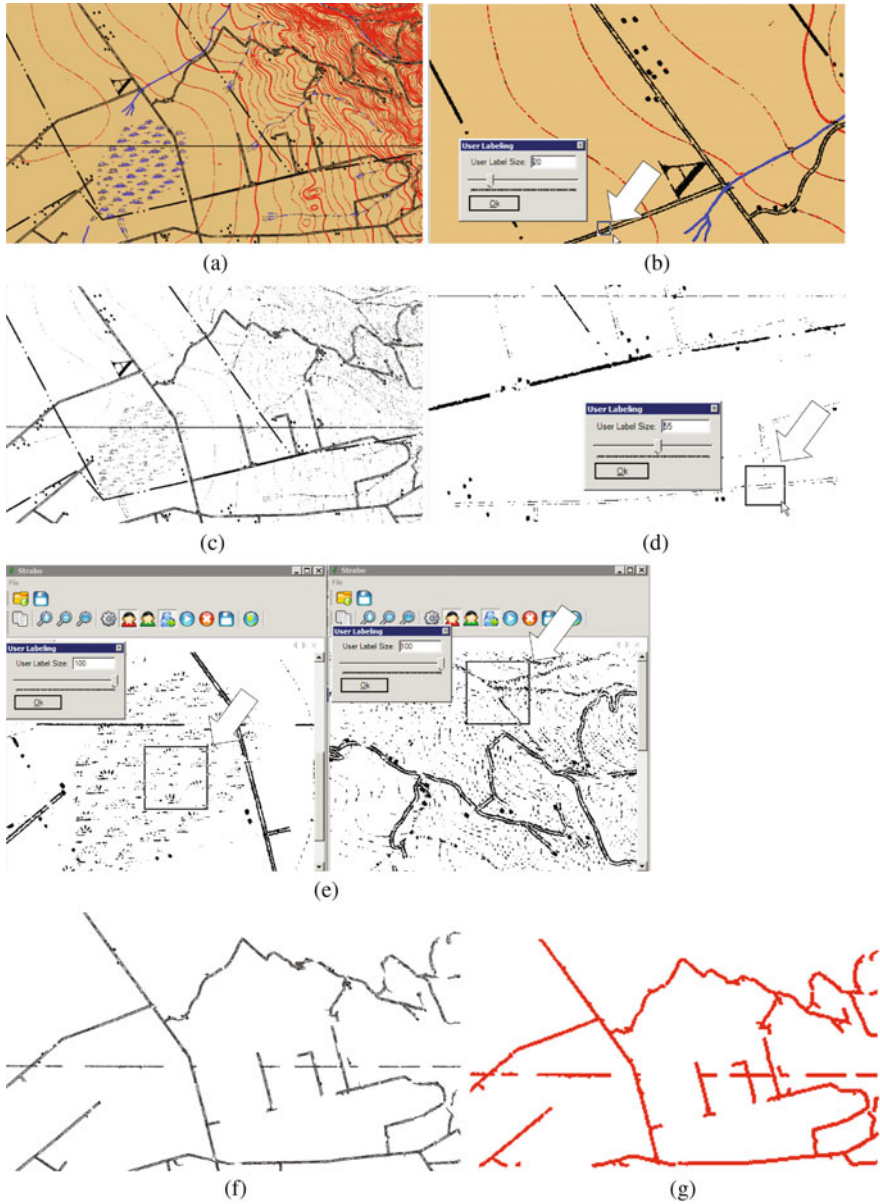


Fig. 3.7 Road layer extraction and cleaning with a few user labeling steps (Figure adapted from [CLK13]). **(a)** An example map after color segmentation. **(b)** A user-provided example of roads. **(c)** The extracted road layer with noise. **(d)** A user-provided example of road pixels. **(e)** A user-provided example of noise objects. **(f)** The cleaned road layer. **(g)** Road vectorization results

3.3.3 *Multi-Model, Context-Based, Automatic Approaches*

Another line of work in the recent development of geographic feature recognition from historical maps uses additional data sources (i.e., contextual data) to help correct recognition errors and improve the level of automation. While this strategy is common in OCR (e.g., using a dictionary), compiling and effectively using a “feature dictionary” for processing historical maps is difficult. Historical maps are typically the only source of geographic information in the past, and finding reference data to help improve the recognition results from an automatic map processing system is a difficult task. For example, a text dictionary built using contemporary data sources does not contain place names that no longer exist. Also, without knowing the map coverage beforehand, multiple dictionary entries can match to a partially recognized label. For example, a partially recognized label “Glas wo” near London could be matched to “Glasgow” when the label is “Glassworks.” Even if map coordinates are known, the map text might not be at the exact location of the geographic features depending on the cartographic labeling practice applied for map production. Weinman [Wei13] presents an approach that overcomes this challenge in recognizing text from historical maps. His approach recognizes text labels in maps to then match the recognized text to a gazetteer by their position patterns using a RANSAC (RANDOM SAmple Consensus) variant, iterative matching algorithm, called MLESAC (Maximum Likelihood Estimation SAmple Consensus) [TZ00]. He showed that this approach could automatically georeference historical maps and improve the recognition accuracy even when the gazetteer only contains 70% of the text in the test maps.

In another work, Chiang et al. [Chi+14] developed a semi-automatic approach that extracts and recognizes text labels in map images in a system called Strabo. While Strabo could achieve over 90% precision and recall in recognizing text labels in scanned contemporary maps, it could only produce 47.6% precision and 83.5% recall on well-conditioned text from historical Ordnance Survey six-inch maps [Chi+16]. Very often only partial labels could be recognized from a historical map (Fig. 3.8a,b) and manual post-processing is required to correct the recognition results. In an effort to test higher levels of automation in text recognition from historical maps in Strabo, Yu, Luo, and Chiang [YLC16] exploit the fact that geographic names for the same area found in different data sources are not independent and use geographic names in OpenStreetMap and other maps covering the same area as the contextual knowledge source. Given a historical map, the task at hand is to recognize all map labels in the map accurately without user intervention. First, the system queries a map repository to find all map editions covering the same area and then extracts and recognizes labels in the identified maps. Second, the system matches the recognized (imperfect) labels using their locations and string similarity using a fuzzy matching algorithm. Finally, the system uses two million geographical names extracted from OpenStreetMap to generate an improved recognition result. For example, by matching “Clture” from the 1935 map to “urch” in the 1900 map, the system finds the word “Church” in the geographic

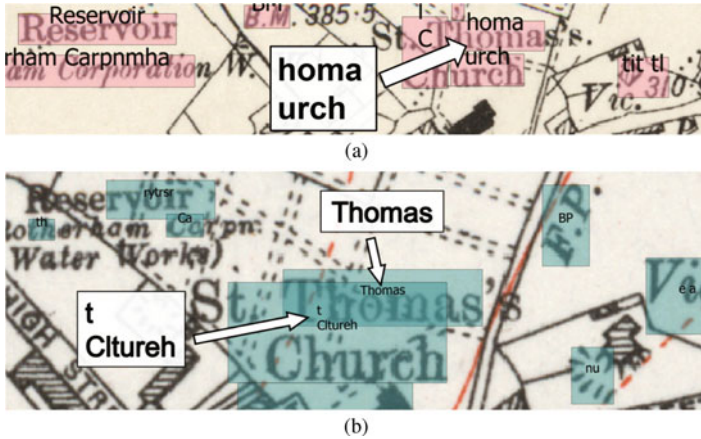


Fig. 3.8 Matching imperfect OCR results from two map editions to improve recognition accuracy (Figure adapted from [YLC16]). (a) Mis-recognizing “St. Thomas Church” as “homa urch” in a 1900 map. (b) Mis-recognizing “St. Thomas Church” as “Thomas t Cltureh” in a 1935 map

names extracted from OpenStreetMap to replace “Cltureh” and “urch” in the 1935 and 1900 recognition results, respectively (Fig. 3.8).

For the multi-model approaches in map text recognition, existing challenges include how to exploit different similarity measures between the extracted map text (which contains recognition errors) and other sources (e.g., gazetteer entries) to 1) prune the search space for finding the matching pattern efficiently and 2) using matches between the OCR text and dictionary entries to learn potential OCR errors specifically for each map type. For example, the character sequences “ni” and “in” is commonly recognized as one character “m” during OCR. With enough training data (matches between OCR text and dictionary entries), the algorithm should be able to learn that the OCR results “Baldwm Hills” is highly likely to be “Baldwin Hills” for a specific map type or condition.

Following the direction of multi-model, context-based map processing, in another effort, Chiang and Leyk [CL15] exploit the fact that geographic information for the same area found in different data sources is not independent to automatically generate training samples and enable fully automatic cartographic symbol recognition. They demonstrated this approach on the recognition of hotel symbols in a map using a gazetteer as the contextual knowledge source. Hotel-related geo-information (presence and locations) found in a map and a gazetteer may not be the same, but they should have some overlap assuming the data are close in time. Given a scanned map covering Baghdad, Iraq (current edition) the task at hand is to find all hotel locations in the map without user intervention (i.e., training the algorithms). First, their system queries GeoNames using the map coordinates and keyword “hotel.” The query results contain two hotel locations (overlapping information between the gazetteer and map): Baghdad Hotel (33.31867, 44.41516) and Palestine Hotel (33.31539, 44.41882). Next, around each of these two locations, the system

cropped a sub-area in the map assuming it contains a hotel symbol. The system then computes feature descriptors from the cropped areas. Finally, the system scans through the entire map to find areas with similar descriptors (matching) and extracts those areas as hotel symbols. In prior work, this descriptor matching process relied on manually selected samples of hotel symbols [CCM14]. In the experiment, this multi-model system extracted 13 hotel locations from the map based on the hotel information from GeoNames. Out of the 13 extracted hotels, 12 are correct (precision 92.3%). There is a total of 17 hotels on the map (recall 70.58%). In previous work on the same task and test data, higher precision and recall (100% and 88.23%, respectively) were only achieved with manually provided hotel samples in the map [CCM14].

In an NSF¹⁶ sponsored project, LinkedMap, the researchers from the University of Southern California and the University of Colorado, Boulder have developed the first version of context-based recognition approaches based on deep machine learning techniques. They use the VGG network (a Very Deep Convolutional Network model) [SZ15] to create DCNNs for feature extraction from scanned maps [Dua+17, Uhl+17, Uhl+18b, Uhl+18a]. For example, they developed a fully automated process for collecting large varieties of high-quality training data using contextual spatial data layers (housing data) for building symbol extraction that overcomes spatial shifts between map features and contextual data [Uhl+17, Uhl+18a] (Fig. 3.9). They have successfully tested fully automated extraction on up to 50 maps using more than one million training samples. However, due to the network architecture limitations, the recognition results are at patch level (in contrast to pixel level) and thus have inaccurate boundaries. To overcome such limitations, they also tested new segmentation methods from computer vision (e.g., [LSD15, Zha+17]) for geographic feature extraction from historical maps [Dua+18]. These methods have achieved significant improvements on non-document images, but their application to maps is difficult because cartographic symbols are significantly different from common (non-document) image objects of interest. Therefore, “weakly” annotated training samples and the reduced spatial resolution during the training of a deep learning model can result in a significant drop of the accuracy in the extraction results.

In sum, multi-model, context-based approaches have the potential to achieve a fully automatic system for map processing using ancillary geographic knowledge to guide the feature sampling and extraction process. This ability to process maps without user intervention is necessary to exploit the full richness of large volume digital historical map archives.

¹⁶National Science Foundation (United States).

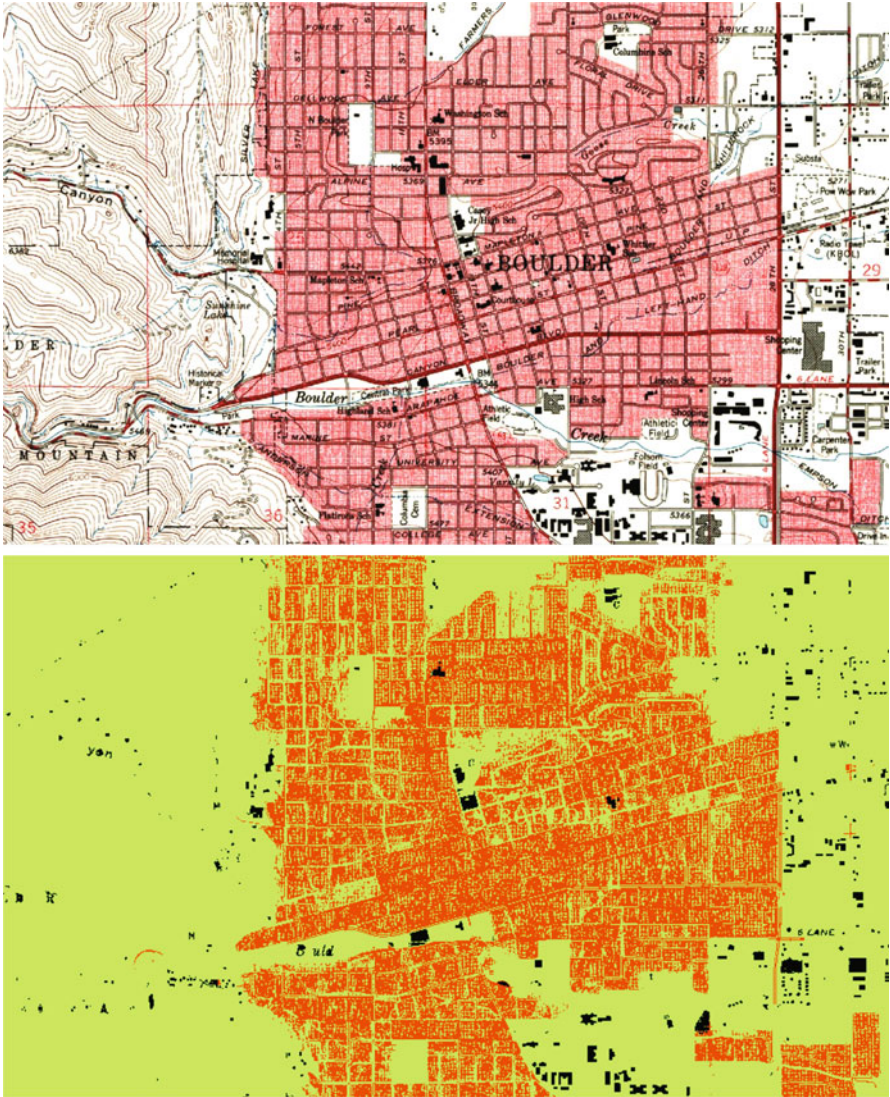


Fig. 3.9 Using contextual data for automatic extraction of settlement features from historical USGS topographic maps (green: no building; black: single building; orange: urban area) (Figure adapted from [Uhl19])

3.4 Modeling and Publishing Map Content

In general, geographic data (as well as the extracted, machine-readable map contents) exists in two major formats: the vector and raster formats.¹⁷ The vector format stores geographic information in one or a series of point locations and sometimes the geometric and topological relationships. Popular vector formats include the Esri Shapefile [ESR98], GeoJSON,¹⁸ and the Open Geospatial Consortium (OGC)¹⁹ Well-Known Text, Well-Known Binary, and KML (Keyhole Markup Language). The Esri Shapefile stores object geometry and their attributes (e.g., location names, addresses). GeoJSON is a lightweight vector format mostly for web GIS and other GIS applications that handle small datasets. Well-Known Text and Well-Known Binary are common data formats for spatial databases (e.g., PostGIS) originated from the OGC. Their definitions are included in the ISO/IEC 13249-3:2016 standard. KML (currently maintained by the OGC) contains not only the object geometries and attributes but also how these geometries should be visualized in an application (e.g., Google Earth and Google Maps Android Software Development Kit).

The vector format does not handle well the geographic phenomena that are continuous over space (e.g., the areas of infill lands extracted from the historical map modeled as a probability surface). In contrast, the raster format can represent continuous phenomena in a grid data structure. The most common raster geographic data type is imagery, including satellite imagery and aerial photos. Every cell in a raster grid contains a value for the location represented by the grid. For example, the pixel (cell) values in a satellite image represent the amount of the reflected sunlight captured by a camera. Other raster geographic data include density maps and heatmaps. Another example of raster data is the extracted map layers from a scanned map in which each pixel represents a probability of the pixel belonging to a specific map layer (e.g., road vs. text layers).

The geographic data in the vector format represent the majority of the existing (machine-readable) structured geographic data. The geographic data in the raster format comprise both structured (e.g., density map where each grid represents a density value) and unstructured geographic data (e.g., scanned historical maps and satellite imagery). The extracted map content can be multi-temporal spatial vector layers (e.g., road centerlines) allowing efficient data handling and analysis, or raster layers in which each raster cell represents how certain it belongs to a particular map feature (e.g., the probability of this cell indicating the urban area in a historical map scan). Vector layers allow a small file size and precise point, skeleton, and boundary representations of map content. Raster layers have the benefit that its data

¹⁷The reader is referred to [Cla10] for a detailed introduction to GISs and GIS data formats.

¹⁸<https://geojson.org/>.

¹⁹<http://www.opengeospatial.org/>.

representation is not limited to a pre-defined boundary and can be easily integrated with other datasets such as historical census data (e.g., census summary tables or reporting area boundaries from the National Historical Geographic Information System, NHGIS [Man+17])²⁰ or land-use and land-cover data (e.g., the USGS Geographic Information Retrieval and Analysis System, GIRAS [Mit77]).

Once the map content is (manually or automatically) converted to machine-readable datasets (either in the vector or raster formats), to make them widely useful and accessible, the challenge is how to add semantic descriptions, include descriptions of uncertainty measures derived from map metadata (rectification and co-registration errors) and the extraction process (e.g., the computational graph of a deep learning model), to the datasets and link the datasets to other sources. Geographic information is already a crucial link connecting entries on the Linked-Data Web [Jan+12]. Also, various ontologies for modeling geographic datasets and spatial relationships exist (e.g., NeoGeo Vocabulary Specification²¹) [Tam18]. Adding semantic descriptions to the map content will enable ontology-supported searches beyond search by place, time, format, and keyword. Representing the extracted dataset as LinkedData [BHB09] could also promote data sharing and support studies that require large historical spatiotemporal datasets. LinkedData represent all of the data as triples, consisting of a subject, predicate, and object using a language called RDF (Resource Description Framework).²² Each element of the triple can be represented as a URI (uniform resource identifier),²³ which provides a unique identifier for the subject, predicate, or object. Objects can also contain a data property such as a string or number. The advantage of representing all of the data as LinkedData is that it provides a standard syntax and terminology enabling a wide range of data representation transformations, and the unique identifiers enable linking across different pieces of metadata and sources.

The challenge here is how to build the techniques that allow a user to model and publish their geographic datasets easily. Existing tools for data integration such as the interactive Web application, Karma,²⁴ has the basic functionality for modeling geometries (i.e., points, lines, and polygons) but not complex or continuous geographic phenomena (e.g., a probability surface). For example, Fig. 3.10 shows that a user can use Karma to map a tabular dataset with geographic coordinates to

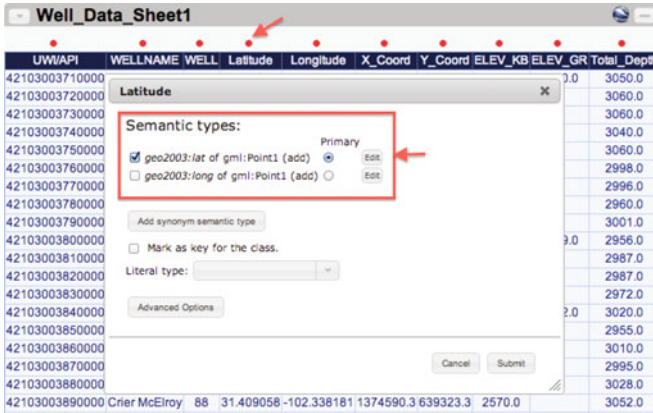
²⁰<https://www.nhgis.org>.

²¹<http://geovocab.org/doc/neogeo>.

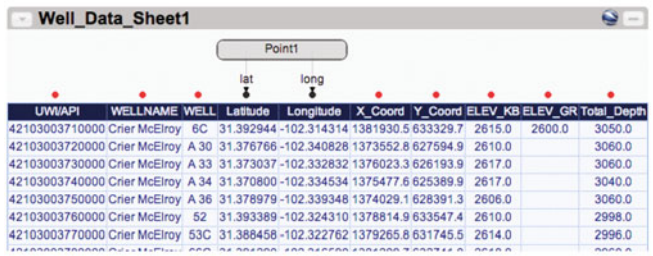
²²<https://www.w3.org/RDF/>.

²³For example, the URI, <https://www.geonames.org/3020251/embrun.html>, refer to the town Embrun in France. The reader is referred to the GeoNames Ontology website (<http://www.geonames.org/ontology/documentation.html>) for more examples about URIs and the Geo Semantic Web.

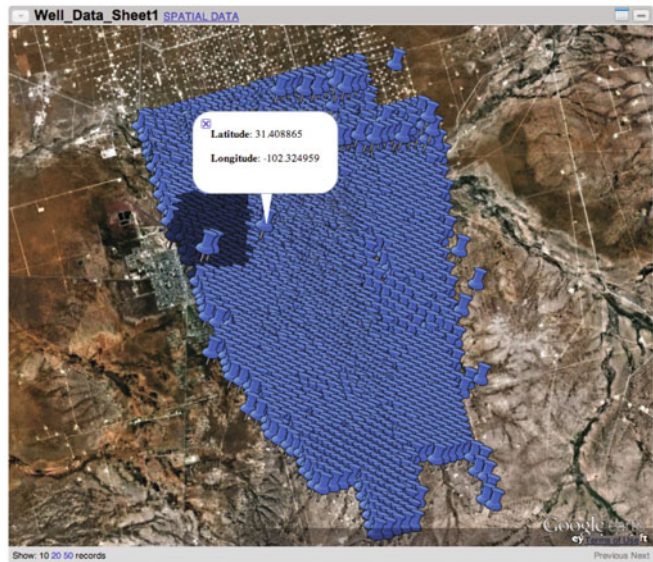
²⁴<http://usc-isi-i2.github.io/karma/>.



(a)



(b)



(c)

Fig. 3.10 Using Karma to map a tabular dataset to the W3C Geospatial Ontology and visualize the dataset in Google Earth. (a) Mapped semantic type of the latitude column in a tabular dataset. (b) Mapped semantic types of both latitude and longitude columns. (c) Display the mapped data on Google Earth

the W3C Geospatial Ontology²⁵ and then visualize the dataset in the Web Google Earth.²⁶ With Karma, the user can also choose to export the modeled geographic datasets to KML or Esri Shapefile. Once the data can be easily modeled with semantic descriptions and linked to other data sources, simply hosting the data on a webpage will greatly help the data to be indexed by the search engines and used by other researchers.

Another notable application for generating linked map data is Recogito,²⁷ which is a Web application developed under the Pelagios Commons.²⁸ In Recogito, the user can upload a map image and provide bounding boxes or point annotations of the text labels on the map. Then the user can search for matching entities in Recogito to provide text annotations and entity linkages of the geographic features on the map. Figure 3.11 shows an example in which the user annotates the text label, “Los Angeles,” and then links the text label to the GeoNames entity of Los Angeles using Recogito. The Web application is easy to use and does not require software installation. Also, since Recogito already contains some geographic datasets (e.g., GeoNames), a simple keyword search allows the user to link the geographic features on a map to entities in other datasets. If the search result does not contain any entity for a keyword, the user can also add new text annotations.

In sum, there exist several data models and ontologies for describing and linking historical information from maps and other historical materials,²⁹ but new tools for either automatically or semi-automatically mapping datasets extracted from maps to the data models still need to be developed. For example, a new technology that automatically generates machine-understandable knowledge about places from historical maps and other sources (e.g., books) and to do so at scale will enable researchers to minimize the time and effort for finding and organizing data for their research. Also, new semi-automatic systems that give data curators the flexibility to define and use a rich semantic model for describing their data, efficiently linking the data across sources, and publishing data online (e.g., a combination of Recogito and Karma) will largely facilitate the process of generating and using machine-understandable data from historical materials.

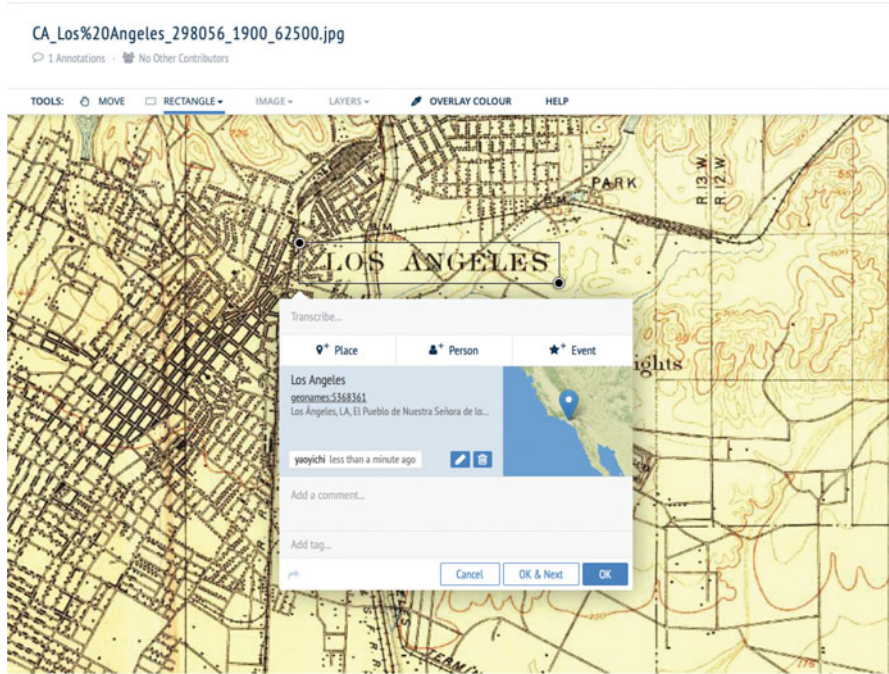
²⁵<https://www.w3.org/2005/Incubator/geo/XGR-geo-ont-20071023/>.

²⁶See the full tutorial here: <https://github.com/usc-isi-i2/Web-Karma/wiki/Working-with-geospatial-data/>.

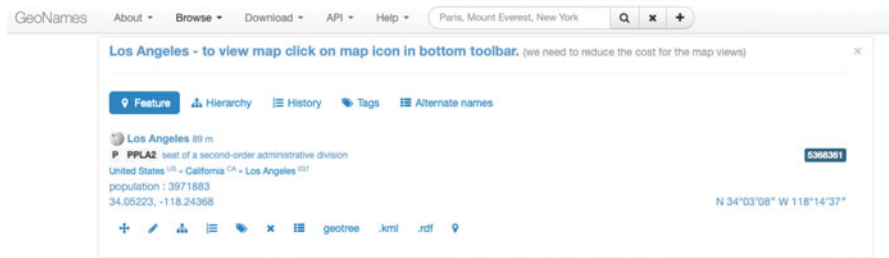
²⁷<https://recogito.pelagios.org/>.

²⁸<http://commons.pelagios.org/>.

²⁹The reader is referred to [Tam18] for an overview of ontologies and the ontologies that describes geographic data.



(a)



(b)

Fig. 3.11 Using Recogito to annotate the place label “Los Angeles” in an USGS historical topographic map and link the label to the GeoNames entity of Los Angeles. (a) Recogito web user interface. (b) The GeoNames feature of Los Angeles (5568321) (source: <https://www.geonames.org/5368361/los-angeles.html>)

3.5 Chapter Summary

This chapter described the challenges and trends for building a map processing, modeling, linking, and publishing framework that enables querying historical map collections as a structured, linked spatiotemporal source. This framework supports answering important questions that require long-term spatiotemporal datasets that

represent states of the Earth's surface in the past. Current challenges include fundamental research in efficient and effective methods for converting map content to machine-readable format, recording provenance and uncertainty information during such digitization processes, modeling map content, provenance, and uncertainty information, and linking the modeled data to other data sources. The resulting tools and datasets will enable a wide utilization of historical maps and support a variety of studies overcoming limitations of current approaches that rely heavily on manual work and are lacking in data sharing.

References

- [Art13] M.G. Arteaga, Historical map polygon and feature extractor, in *MapInteract 2013, Proceedings of the 1st ACM SIGSPATIAL International Workshop on MapInteraction, November 5th, 2013, Orlando, Florida, USA* (2013), pp. 66–71. <https://doi.org/10.1145/2534931.2534932>
- [Bar+18] B. Barz, T.C. van Dijk, B. Spaan, J. Denzler, Putting user reputation on the map: unsupervised quality control for crowdsourced historical data, in *Proceedings of the 2nd ACM SIGSPATIAL Workshop on Geospatial Humanities, GeoHumanities'18* (ACM, New York, 2018), pp. 3:1–3:6. ISBN: 978-1-4503-6032-6. <https://doi.org/10.1145/3282933.3282937>
- [Bas+18] F. Bastani, S. He, S. Abbar, M. Alizadeh, H. Balakrishnan, S. Chawla, S. Madden, Machine-assisted map editing, in *Proceedings of the 26th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, SIGSPATIAL '18* (ACM, New York, 2018), pp. 23–32. ISBN: 978-1-4503-5889-7. <https://doi.org/10.1145/3274895.3274927>
- [BD15] B. Budig, T.C. van Dijk, Active learning for classifying template matches in historical maps, in *Discovery Science*, ed. by N. Japkowicz, S. Matwin. Lecture Notes in Computer Science (Springer, Berlin, 2015), pp. 33–47. ISBN: 9783319242811, 9783319242828. https://doi.org/10.1007/978-3-319-24282-8_5
- [BDW16] B. Budig, T.C.V. Dijk, A. Wolff, Matching labels and markers in historical maps: an algorithm with interactive postprocessing, in *ACM Transactions on Spatial Algorithms and Systems (TSAS) 2.4* (2016), pp. 13:1–13:24. ISSN: 2374-0353. <https://doi.org/10.1145/2994598>
- [Bea14] C.S. Beattie, 3D visualization models as a tool for reconstructing the historical landscape of the Ballona creek watershed, MA thesis, University of Southern California, 2014
- [BHB09] C. Bizer, T. Heath, T. Berners-Lee, Linked data - the story so far. *Int. J. Semant. Web Inf. Syst.* **5**(3), 1–22 (2009). ISSN: 1552-6283
- [Bud+16] B. Budig, T.C. van Dijk, F. Feitsch, M.G. Arteaga, Polygon consensus: smart crowdsourcing for extracting building footprints from historical maps, in *Proceedings of the 24th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, SIGSPACIAL '16* (ACM, New York, 2016), pp. 66:1–66:4. ISBN: 978-1-4503-4589-7. <https://doi.org/10.1145/2996913.2996951>
- [Bud16] B. Budig, Efficient algorithms and user interaction for metadata extraction from historical maps, in *Proceedings of the 2Nd ACM SIGSPATIAL PhD Workshop, SIGSPATIAL PhD '15* (ACM, New York, 2016), pp. 4:1–4:4. ISBN: 978-1-4503-3980-3. <https://doi.org/10.1145/2855680.2855841>
- [Bud18] B. Budig, Extracting spatial information from historical maps: algorithms and interaction, PhD thesis, University of Würzburg, 2018

- [BvK16] B. Budig, T.C. van Dijk, F. Kirchner, Glyph miner: a system for efficiently extracting glyphs from early prints in the context of OCR, in *2016 IEEE/ACM Joint Conference on Digital Libraries (JCDL)* (2016), pp. 31–34
- [CCM14] Y.-Y. Chiang, P. Chioh, S. Moghaddam, A training-by-example approach for symbol spotting from raster maps, in *Proceedings of the 8th International Conference on Geographic Information Science* (2014), pp. 264–269
- [Che+04] C.-C. Chen, C.A. Knoblock, C. Shahabi, Y.-Y. Chiang, S. Thakkar, Automatically and accurately conflating orthoimagery and street maps, in *Proceedings of the 12th Annual ACM International Workshop on Geographic Information Systems, GIS '04* (ACM, New York, 2004), pp. 47–56. ISBN: 9781581139792. <https://doi.org/10.1145/1032222.1032231>
- [Chi+14] Y.-Y. Chiang, S. Moghaddam, S. Gupta, R. Fernandes, C.A. Knoblock, From map images to geographic names, in *Proceedings of the 22nd ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems* (ACM, New York, 2014), pp. 581–584. ISBN: 9781450331319. <https://doi.org/10.1145/2666310.2666374>
- [Chi+15] Y.-Y. Chiang, S. Leyk, N.H. Nazari, S. Moghaddam, The impact of graphical quality on automatic text recognition in digital maps, in *Proceedings of the 27th International Cartographic Conference* (2015) ISBN: 9788588783119
- [Chi+16] Y.-Y. Chiang, S. Leyk, N.H. Nazari, S. Moghaddam, T.X. Tan, Assessing the impact of graphical quality on automatic text recognition in digital maps. *Comput. Geosci.* **93**, 21–35 (2016). ISSN: 0098-3004. <https://doi.org/10.1016/j.cageo.2016.04.013>
- [Chi10] Y.-Y. Chiang, Harvesting geographic features from heterogeneous raster maps, PhD thesis, Los Angeles, CA, USA: University of Southern California, 2010. ISBN: 9781124412498
- [Chi13] Y.-Y. Chiang, Strabo: a complete system for label recognition in maps, in *Proceedings of the 26th International Cartographic Conference (ICC'13)* (2013), pp. 838–838. ISBN: 9781907075063
- [Chi15] Y.-Y. Chiang, Querying historical maps as a unified, structured, and linked spatiotemporal source: vision paper, in *Proceedings of the 23rd SIGSPATIAL International Conference on Advances in Geographic Information Systems. GIS '15* (ACM, New York, 2015), pp. 16:1–16:4. ISBN: 9781450339674. <https://doi.org/10.1145/2820783.2820887>
- [Chi17] Y.-Y. Chiang, Unlocking textual content from historical maps - potentials and applications, trends, and outlooks, in *Recent Trends in Image Processing and Pattern Recognition*, ed. by K. Santosh, M. Hangarge, V. Bevilacqua, A. Negi (Springer, Singapore, 2017), pp. 111–124. ISBN: 978-981-10-4859-3
- [CK08] Y.-Y. Chiang, C.A. Knoblock, Automatic extraction of road intersection position, connectivity, and orientations from raster maps, in *Proceedings of the 16th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems* (ACM Press, New York, 2008), pp. 1–10. ISBN: 9781605583235. <https://doi.org/10.1145/1463434.1463463>
- [CK14] Y.-Y. Chiang, C.A. Knoblock, Recognizing text in raster maps. *GeoInformatica* **19**(1), 1–27 (2014). ISSN: 1384-6175, 1573-7624. <https://doi.org/10.1007/s10707-014-0203-9>
- [CL15] Y.-Y. Chiang, S. Leyk, Exploiting online gazetteer for fully automatic extraction of cartographic symbols, in *Proceedings of the 27th International Cartographic Conference* (2015). ISBN: 9788588783119
- [Cla10] K.C. Clarke, *Getting Started with Geographic Information Systems* (Pearson, London, 2010). ISBN: 9780131494985
- [CLK13] Y.-Y. Chiang, S. Leyk, C.A. Knoblock, Efficient and robust graphics recognition from historical maps, in *Graphics Recognition. New Trends and Challenges: 9th International Workshop, GREC 2011, Seoul, Korea, September 15–16, 2011, Revised Selected Papers*, vol. 7423, ed. by Y.-B. Kwon, J.-M. Ogier. Lecture Notes in Computer Science, GREC'11 (Springer, Berlin, 2013), pp. 25–35. ISBN: 9783642368233. https://doi.org/10.1007/978-3-642-36824-0_3

- [CLK14] Y.-Y. Chiang, S. Leyk, C.A. Knoblock, A survey of digital map processing techniques, in *ACM Comput. Surv.* **47**(1), 1–44 (2014). ISSN: 0360-0300. <https://doi.org/10.1145/2557423>
- [Dua+17] W. Duan, Y.-Y. Chiang, C.A. Knoblock, V. Jain, D. Feldman, J.H. Uhl, S. Leyk, Automatic alignment of geographic features in contemporary vector data and historical maps, in *Proceedings of the 1st Workshop on Artificial Intelligence and Deep Learning for Geographic Knowledge Discovery* (ACM, New York, 2017), pp. 45–54
- [Dua+18] W. Duan, Y. Chiang, C.A. Knoblock, S. Leyk, J. Uhl, Automatic generation of precisely delineated geographic features from georeferenced historical maps using deep learning, in *Proceedings of the AutoCarto* (2018)
- [ESR98] ESRI, ESRI shapefile technical description, Tech. rep. ESRI, 1998
- [FC97] S. Frischknecht, A. Carosio, Raster-based methods to extract structured information from scanned topographic maps, in *International Archives of Photogrammetry and Remote Sensing*, vol. 32, Part 3-4W2 (1997), pp. 1–5
- [FK97] S. Frischknecht, E. Kanani, Automatic interpretation of scanned topographic maps: a raster-based approach, in *Graphics Recognition Algorithms and Systems*, vol. 1398 (Springer, Berlin, 1997), pp. 207–220. ISBN: 9783540643814. https://doi.org/10.1007/3-540-64381-8_50
- [GE07] I.N. Gregory, P.S. Ell, *Historical GIS: Technologies, Methodologies, and Scholarship*, vol. 39. Cambridge Studies in Historical Geography (Cambridge University Press, Cambridge, 2007). ISBN: 9781139467711
- [GE15] B. Godfrey, H. Eveleth, An adaptable approach for generating vector features from scanned historical thematic maps using image enhancement and remote sensing techniques in a geographic information system. *J. Map Geogr. Libr.*, 18–36 (2015). ISSN: 1542-0353. <https://doi.org/10.1080/15420353.2014.1001107>
- [Gel08] J. Gelernter, MapSearch: a protocol and prototype application to find maps, PhD thesis. Rutgers, The State University of New Jersey, 2008
- [GGH15] D. Garijo, Y. Gil, A. Harth, Challenges for provenance analytics over geospatial data, in *Provenance and Annotation of Data and Processes*, vol. 8628, ed. by B. Ludäscher, B. Plale. Lecture Notes in Computer Science (Springer, Berlin, 2015), pp. 261–263
- [GS62] C.R. Greenwalt, M.E. Shultz, Principles of error theory and cartographic applications, Tech. rep. 1962
- [Jan+12] K. Janowicz, S. Scheider, T. Pehle, G. Hart, Geospatial semantics and linked spatiotemporal data – past, present, and future. *Semantic Web* **3**(4), 321–332 (2012). <https://doi.org/10.3233/SW-2012-0077>
- [Kur13] L. Kurashige, Rethinking anti-immigrant racism: lessons from the los angeles vote on the 1920 Alien land law. *South. Calif. Q.* **95**(3), 265–283 (2013). ISSN: 0038-3929. <https://doi.org/10.1525/scq.2013.95.3.265>
- [KZ03] A. Khotanzad, E. Zink, Contour line and geographic feature extraction from USGS color topographical paper maps. *IEEE Trans. Pattern Anal. Mach. Intell.* **25**(1), 18–31 (2003). Issn: 0162-8828. <https://doi.org/10.1109/TPAMI.2003.1159943>
- [LBW05] S. Leyk, R. Boesch, R. Weibel, A conceptual framework for uncertainty investigation in map-based land cover change modelling. *Trans. GIS* **9**(3), 291–322 (2005). <https://doi.org/10.1111/j.1467-9671.2005.00220.x> eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1467-9671.2005.00220.x>
- [LC18a] H. Lin, Y.-Y. Chiang, An uncertainty aware method for geographic data conflation, in *Proceedings of the 7th ACM SIGSPATIAL International Workshop on Analytics for Big Geospatial Data, BigSpatial 2018* (ACM, Seattle, 2018), pp. 20–27. ISBN: 978-1-4503-6041-8. <https://doi.org/10.1145/3282834.3282842>
- [LC18b] H. Lin, Y.-Y. Chiang, SRC: automatic extraction of phrase-level map labels from historical maps. *SIGSPATIAL Spec.* **9**(3), 14–15 (2018)
- [LLZ18] H. Li, J. Liu, X. Zhou, Intelligent map reader: a framework for topographic map understanding with deep learning and gazetteer. *IEEE Access* **6**, 25363–25376 (2018). ISSN: 2169-3536. <https://doi.org/10.1109/ACCESS.2018.2823501>

- [LSD15] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2015), pp. 3431–3440
- [Man+17] S. Manson, J. Schroeder, D. Van Riper, S. Ruggles, et al., IPUMS National Historical Geographic Information System: Version 12.0 [Database], Minneapolis: University of Minnesota (2017)
- [Mit77] W.B. Mitchell, GIRAS: a geographic information retrieval and analysis system for handling land use and land cover data, 1059. US Govt. Print. Off. (1977)
- [Nag+97] G. Nagy, A. Samal, S. Seth, T. Fisher, et al., Reading street names from maps-technical challenges, in *Proceedings of GIS/LIS* (1997)
- [Pag+99] L. Page, S. Brin, R. Motwani, T. Winograd, The PageRank citation ranking: bringing order to the web, Tech. rep. Stanford InfoLab, 1999
- [Pez11] A. Pezeshk, Feature extraction and text recognition from scanned color topographic maps, PhD thesis, Pennsylvania State University, 2011
- [S+15] R. Simon, E. Barker, L. Isaksen, et al., Linking early geospatial documents, one place at a time: annotation of geographic documents with recogito, in e- (2015). <http://oro.open.ac.uk/43613/>
- [Sim+10] R. Simon, C. Sadilek, J. Korb, M. Baldauf, B. Haslhofer, Tag clouds and old maps: annotations as linked spatiotemporal data in the cultural heritage domain, in *Workshop On Linked Spatiotemporal Data, Zurich, Switzerland* (2010)
- [Sim+14] R. Simon, P. Pilgerstorfer, L. Isaksen, E. Barker, Towards semi-automatic annotation of toponyms on old maps. e - Perimetreon 9(3), 105–128 (2014)
- [SZ15] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7–9, 2015, Conference Track Proceedings* (2015)
- [Tam18] T. Tambassi, *The Philosophy of Geo-Ontologies* (Springer, Berlin, 2018)
- [TZ00] P.H.S. Torr, A. Zisserman, MLESAC: a new robust estimator with application to estimating image geometry. Comp. Vision Image Underst. 78(1), 138–156 (2000). ISSN: 1077-3142. <https://doi.org/10.1006/cviu.1999.0832>
- [Uhl+17] J.H. Uhl, S. Leyk, Y.-Y. Chiang, W. Duan, C.A. Knoblock, Extracting human settlement footprint from historical topographic map series using context-based machine learning, in *IET Conference Proceedings* (2017)
- [Uhl+18a] J.H. Uhl, S. Leyk, Y.-Y. Chiang, W. Duan, C.A. Knoblock, Spatialising uncertainty in image segmentation using weakly supervised convolutional neural networks: a case study from historical map processing. IET Image Process. 12(11), 2084–2091 (2018)
- [Uhl+18b] J. Uhl, S. Leyk, Y.-Y. Chiang, W. Duan, C. Knoblock, Map archive mining: visual-analytical approaches to explore large historical map collections. ISPRS Int. J. Geo-Inform. 7(4), 148 (2018)
- [Uhl19] J.H. Uhl, Spatio-temporal information extraction under uncertainty using multi-source data integration and machine learning: applications to human settlement modelling, PhD thesis, University of Colorado (2019)
- [Wei13] J. Weinman, Toponym recognition in historical maps by gazetteer alignment, in *Proceedings of the 12th International Conference on Document Analysis and Recognition* (2013), pp. 1044–1048. <https://doi.org/10.1109/ICDAR.2013.209>
- [Wei17] J. Weinman, Geographic and style models for historical map alignment and toponym recognition, in *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, vol. 1 (IEEE, Piscataway, 2017), pp. 957–964
- [YLC16] R. Yu, Z. Luo, Y.-Y. Chiang, Recognizing text in historical maps using maps from multiple time periods, in *2016 23rd International Conference on Pattern Recognition (ICPR)* (IEEE, Piscataway, 2016), pp. 3993–3998
- [Zha+17] H. Zhao, J. Shi, X. Qi, X. Wang, J. Jia, Pyramid scene parsing network, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2017), pp. 2881–2890

Chapter 4

Training Deep Learning Models for Geographic Feature Recognition from Historical Maps



Abstract Historical map scans contain valuable information (e.g., historical locations of roads, buildings) enabling the analyses that require long-term historical data of the natural and built environment. Many online archives now provide public access to a large number of historical map scans, such as the historical USGS (United States Geological Survey) topographic archive and the historical Ordnance Survey maps in the United Kingdom. Efficiently extracting information from these map scans remains a challenging task, which is typically achieved by manually digitizing the map content. In computer vision, the process of detecting and extracting the precise locations of objects from images is called semantic segmentation. Semantic segmentation processes take an image as input and classify each pixel of the image to an object class of interest. Machine learning models for semantic segmentation have been progressing rapidly with the emergence of Deep Convolutional Neural Networks (DCNNs or CNNs). A key factor for the success of CNNs is the wide availability of large amounts of (labeled) training data, but these training data are mostly for daily images not for historical (or any) maps. Today, generating training data needs a significant amount of manual labor that is often impractical for the application of historical map processing. One solution to the problem of training data scarcity is by transferring knowledge learned from a domain with a sufficient amount of labeled data to another domain lacking labeled data (i.e., transfer learning). This chapter presents an overview of deep-learning semantic segmentation models and discusses their strengths and weaknesses concerning geographic feature recognition from historical map scans. The chapter also examines a number of transfer learning strategies that can reuse the state-of-the-art CNN models trained from the publicly available training datasets for the task of recognizing geographic features from historical maps. Finally, this chapter presents a comprehensive experiment for extracting railroad features from USGS historical topographic maps as a case study.

4.1 Introduction

Historical maps store valuable information documenting human activities and natural features on Earth over long periods of time.¹ For example, the USGS (United States Geological Survey) historical topographic map archive records the evolution of railroad networks in the USA dating back to the early 1900s or even the late 1800s. Such information is essential for the studies that require detailed, long-term historical geographic data. With hundreds of thousands of historical maps scanned and stored in digital archives, existing digital map processing methods that need user interventions for the recognition of map content remain inefficient for processing large numbers of maps [CLK14]. For example, Chiang and Knoblock [CK13] developed a semi-automatic approach for extracting road vectors from maps. Their approach requires users to label a few samples of road lines and road intersections. Samet and Hancer [SH12] built a semi-automatic system to reconstruct contour lines from topographic map scans, in which the user selects the dimension of a “mask” indicating the gaps between the extracted broken contour lines for reconnecting them. In addition, some systems use prior knowledge to separate map layers. For example, Henderson et al. [Hen+09] uses the color index of USGS topographic maps to separately extract each map layer. *These systems do not scale well for processing large numbers and varieties of historical maps.*

In recent years, Deep Convolutional Neural Networks (DCNNs or CNNs) have shown promising performance in many computer vision tasks when a sufficient amount of labeled training data are available (e.g., [SZ15, He+16, Hua+17]). One of the main advantages of CNNs is their capability of handling large diversified input data (e.g., face recognition from images of various lighting conditions and background). To train CNNs for a number of image understanding tasks (e.g., face recognition, street scenes parsing), many publicly labeled datasets are available, including ImageNet [Rus+15b], PASCAL [EW11], and CamVid [Bro+08]. As an example, ImageNet² contains 14,197,122 images describing 21,841 concepts (e.g., “wheeled vehicle”) in WordNet [Fel98]. Training on ImageNet, the state-of-the-art CNNs have achieved impressively low error rates in image recognition challenges. For example, VGG [SZ15], GoogLeNet [Sze+15], and ResNet [He+16] have achieved 7.3%, 6.7%, and 3.57% error rates in the ImageNet Challenges in image classification, detection, and localization in 2014, 2014, and 2015, respectively [Rus+15a]. However, (scanned) maps are a special type of document image, and thus, training and using a deep learning model for extracting geographic features requires careful considerations of:

- how to select an appropriate CNN architecture for extracting accurate boundaries of geographic features and

¹In this chapter, the term “historical maps” refer to the professionally prepared maps by cartographers and typically published by government mapping agencies. The term “map scan” refers to scanned images of maps.

²<http://www.image-net.org>.

- how to reuse, if possible, the learned knowledge from the existing state-of-the-art CNNs trained with daily images to help improve the performance of identifying geographic features from map scans.

Without an understanding of these issues as well as the best practice to handle them, unsuccessful attempts in using deep learning models for automatic understanding map scans could slow down the further advancement of digital map processing techniques and their applications. In many other application domains (e.g., autonomous driving) that use CNNs or other deep learning models, comprehensive reviews of deep learning technologies concerning these domains are available (e.g., [Bal+19]) but not for digital map processing, which is unfortunate given the huge amount of available historical map scans, their valuable content, and the wide availability of CNNs. This chapter attempts to provide a clear connection between deep learning technologies and the domain of digital map processing for building an automatic geographic feature recognition system using CNNs. Specifically, this chapter presents an overview and suggestions of the best practice for using deep-learning semantic segmentation models exploiting a transfer learning strategy for recognizing geographic features from historical map scans.

The remainder of this chapter is structured as follows. Section 4.2 describes the challenges of using deep learning for extracting geographic features from historical map scans. Section 4.3 provides a brief background on CNNs and the state-of-the-art semantic segmentation models concerning map processing. Section 4.4 introduces the concept and strategies of transfer learning for reusing pre-trained deep learning models for recognizing geographic features from historical map scans. Section 4.5 presents a comprehensive experiment on feature recognition from historical maps scans using various deep learning models and transfer learning strategies. Section 4.6 summarizes the chapter and discusses future work.

4.2 Challenges in Using CNNs on Historical Maps

In the field of machine learning, artificial neural networks (ANNs) refer to a computational graph that can be used for learning a model to describe complex relationships within a set of data (unsupervised learning) or between a set of input data and their output from a process (supervised learning) as well as for other learning tasks (e.g., reinforcement learning). The computational graph (i.e., the network architecture) contains a set of nodes (neurons) and their connections (synapse). A Convolutional Neural Network (CNN) (or Convolutional Neural Networks, CNNs) is a special type of ANNs that have a deep network structure with a feed-forward information propagation strategy (i.e., network without cycles). In practice, CNNs are used in a variety of image recognition tasks, including object detection, recognition, and segmentation. Extracting geographic features (or

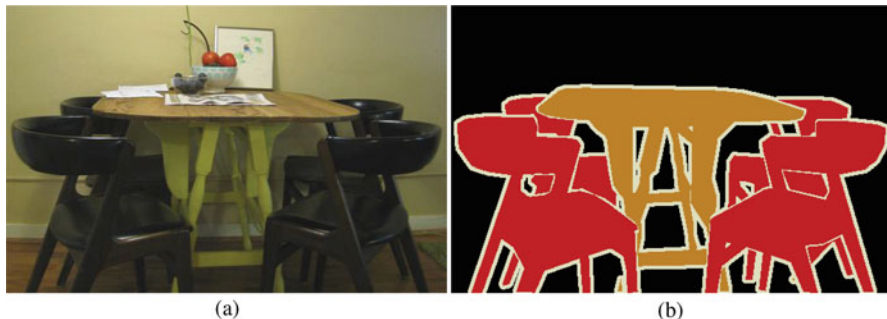


Fig. 4.1 Example training data from the PASCAL VOC2011 dataset. (a) An example training image. (b) The corresponding pixel-wise labeled segmentation image, including the background (black), table (orange), and chairs (red)

delineating their pixel locations)³ from map scans can be seen as the computer vision task of “semantic segmentation.” CNNs for semantic segmentation aim to assign each pixel in an image to an object class (e.g., cars, people, ground, or sky). In particular, for training and testing machine learning models for semantic segmentation, the 2012 PASCAL Visual Object Classes (VOC) dataset [Eve+]⁴ contains 20 segmentation classes (e.g., object classes of tables, chairs, and people) and 6929 labeled image segmentation results (Fig. 4.1).

Many of the state-of-the-art CNNs use the learned “feature representations” from an existing, pre-trained CNN and then replace some or add additional network structures and train the updated CNN with additional data for a different recognition task (from the recognition task of the original CNN). For example, a CNN for image classification (e.g., ResNet [He+16]) can be used to generate feature representations in another CNN for detecting scene text (e.g., EAST [Zho+17]); a CNN for identifying boats from satellite imagery [Yan+18] can also be extended for detecting scene text [Jia+17]. Some of these modifications require significant changes in the model architecture (i.e., network structures), and others only need to retrain some layers of the network without modifying the architecture. The latter type of modification is called “transfer learning,” which is especially useful when training data is scarce.

This section presents two major challenges towards applying semantic segmentation CNNs to historical map scans for automatically extracting geographic features.

³In this chapter, extracting or recognizing geographic features refers to the process of annotating each pixel in a map scan with a class label of a type of geographic feature (e.g., roads, text labels, or buildings).

⁴<http://host.robots.ox.ac.uk/pascal/VOC/index.html>.

4.2.1 *Accurate Boundary Delineation of Geographic Features*

Most of the existing semantic segmentation models trained and tested on the public datasets, such as the PASCAL VOC, may not be suitable directly for extracting accurate boundaries of the geographic features in historical map archives [Dua+18]. In recent work, the Very Deep Convolutional Networks (VGG) [SZ15] were used for feature extraction from scanned historical maps, but due to the architecture limitations, the recognition results could have inaccurate and shifted boundaries [Dua+18, Uhl+17]. This is because VGG, like many CNNs, abstracts the input images through multiple down-sampling processes (e.g., the convolutional and max-pooling layers), which can result in the loss of local image information and poorly delineated object contours or boundaries. Figure 4.2 shows examples of semantic segmentation results from a number of deep learning models where small objects in the input image (e.g., light poles) are often not recognized and irregular contours are often generalized in the results. For many computer vision applications, well-recognized object boundaries are often not a strict requirement (e.g., for recognizing individual persons in an image). Yet, for the extracted map features to be useful in scientific studies, the delineated boundaries need to be precise given that a one-pixel offset in the feature geometry may correspond to a distance of several meters on the earth surface.

Maps, as a special type of document image, contain cartographic symbols with boundary representations significantly different from the (non-document) image objects in the public datasets commonly used for training and testing CNNs for semantic segmentation models. Compared to images in the public datasets, the image pixels representing a geographic feature of interest in a map document occupy only a small proportion of the entire image (Fig. 4.3b). Therefore, even slightly reducing the spatial resolution during the training of CNNs can result in a significant decrease of the spatial accuracy in the extraction results. In addition, the graphical representations of cartographic symbols (e.g., colors and shapes) belonging to different map layers can be very similar when the focal window is small, which can result in high proportions of false positives in the extraction results. For example, Fig. 4.4 shows waterline symbols and their corresponding text. If only a small portion of the characters (e.g., the bottom part of the character “e”) is considered during the semantic segmentation process, it can be misclassified as a part of the waterline symbols. These challenges are often not a major concern when deep learning models are used for the task of recognizing objects in daily images (e.g., segmenting hair areas instead of individual hairs from an image).

Generating accurate object boundaries is still an open research topic in semantic segmentation, and new methods have been proposed to overcome such limitations of inaccurate delineated boundaries resulting from image down-sampling, (e.g., [LSD15, YK16, BST15]) and achieved significant improvements on non-document images (e.g., daily images in the PASCAL-Context). Section 4.3 will present an overview of common CNN architectures and recommendations for feature recognition from map scans (Fig. 4.4).

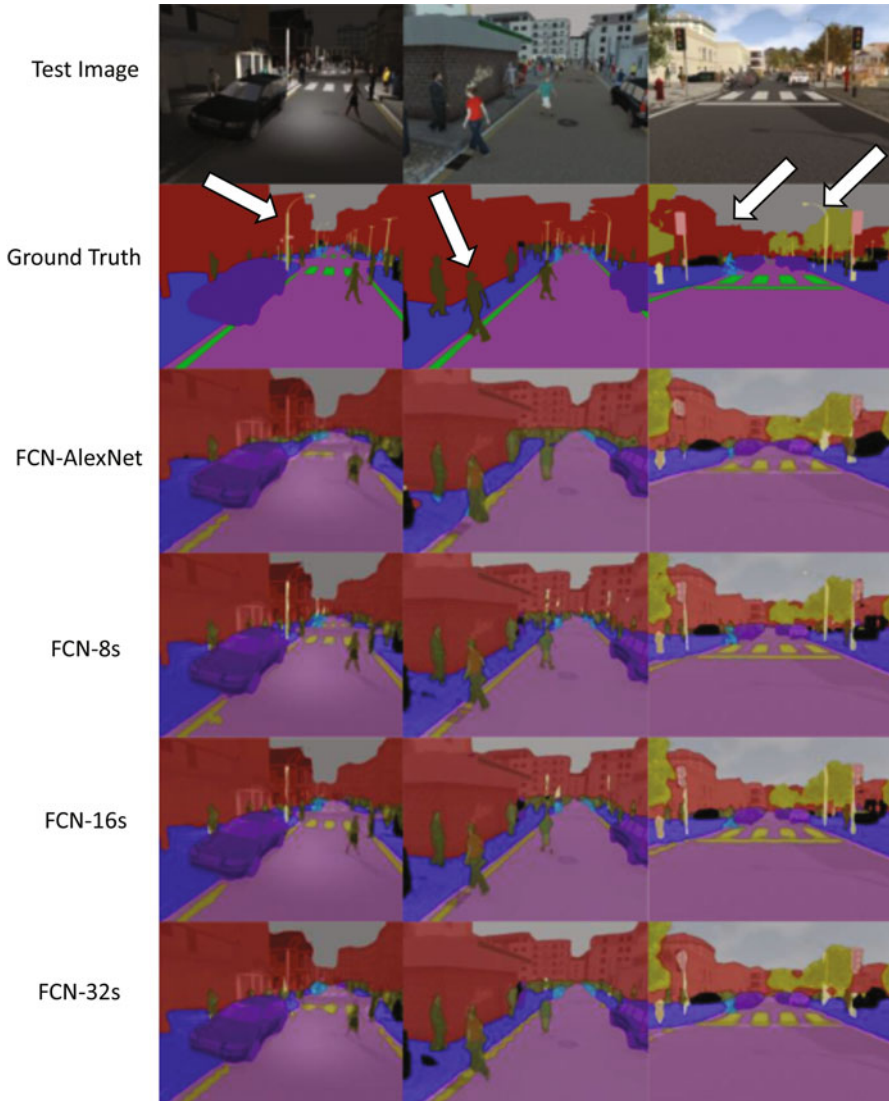


Fig. 4.2 Example semantic segmentation results from FCN variants: the white arrows indicate example image objects that are either small objects or small contour details, which are difficult to recognize. (Figure adapted from [KU19])

4.2.2 Scarce Training Data for Cartographic Documents

The second challenge is that semantic segmentation models trained with the publicly available, labeled datasets do not work well for (historical) maps without a sufficient amount of labeled training data from map scans. This is because the training and

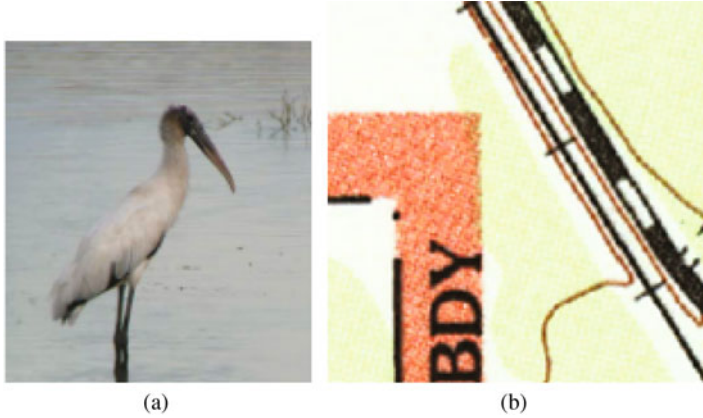


Fig. 4.3 The object-of-interest (a bird) in typical training datasets for semantic segmentation models occupies a large region in the image (a) in contrast to the typical geographic feature (railroads) in maps (b) about 2.5% of the entire image

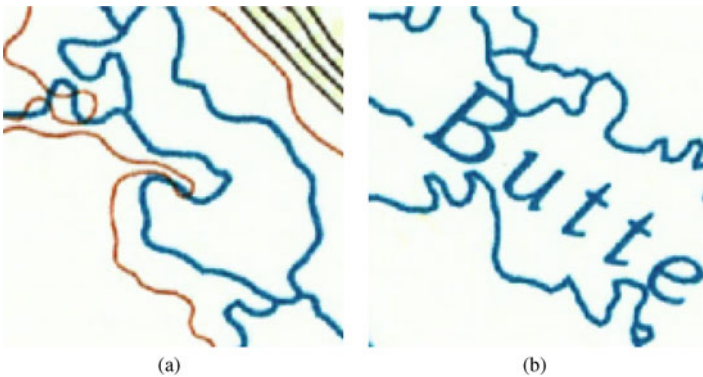
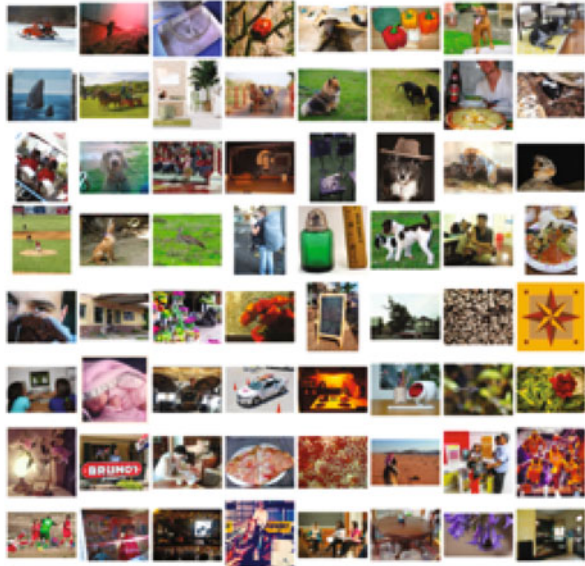


Fig. 4.4 The graphical representations of cartographic symbols belonging to different map layers can be very similar, e.g., waterlines (a) and characters (b)

target data are from very different data domains. Most labeled training data from public datasets are daily images containing real-world objects, while maps are document images with geographic features represented by cartographic symbols. Figure 4.5 shows image samples from ImageNet, a public dataset commonly used for training CNNs and a USGS topographic historical map. Generating labeled data for training semantic segmentation models requires a significant amount of resources, which is often impractical to find for the domain of map processing. In contrast, other application domains, such as the recognition of traffic signs and cars from photos, have drawn large numbers of contributions of labeled data from both the industry and academia.



(a)



(b)

Fig. 4.5 Image samples form ImageNet and the USGS topographic historical map archive. **(a)** Samples of training data from ImageNet. **(b)** A historical USGS topographic map (Los Angeles, 1:24,000, circa 1966)

A solution to overcome the problem of insufficient labeled data is the technique of transfer learning, which takes a machine learning model trained on some datasets and further trains the model with datasets from a new (often different) data domain so that the final model can be applied to the new data domain (sometimes called “fine-tuning” in the deep learning literature). The goal of transfer learning is, with only scarce labeled data, to achieve the recognition performance as if there is sufficient training data by transferring the knowledge learned from one domain with sufficient labeled data to another domain. For example, assuming we have a large number of labeled data for dogs in daily images and a small number of labeled data for cats, also in daily images. We can first train a CNN model to recognize dogs in the images. With a sufficient amount of training data, this CNN model will learn a number of representative image features and their combinations for recognizing dogs in the images. Then we can use transfer learning techniques to fine tune the model with labeled cat data for recognizing cats in the images.

Many researchers (e.g., [ALL17, Mar+16, Cas+15]) have taken advantage of transfer learning for their segmentation tasks when the available training data is limited. Audebert et al. [ALL17] investigated the performance of several deep fully convolutional neural networks (DFCNNs) for the segmentation of Earth Observation images (remote sensing data over an urban area). They successfully transferred a CNN trained with daily images (ImageNet) to handle remote sensing images. Marmanis et al. [Mar+16] built a two-stage CNN classifier to recognize urban areas in remote sensing images. In the first stage, their network derives new data representations using a CNN trained with ImageNet. In the second stage, the network uses the representations derived from the first stage as the input to recognize urban areas in remote sensing images. Compared with the best results of the same task from previous work, their two-stage CNN classifier has improved the recognition accuracy from 83.1 to 92.4%.

Castelluccio et al. [Cas+15] tested three training strategies using two state-of-the-art deep learning models (CaffeNet and GoogLeNet) for semantic land-use classification from remote sensing scenes. The first strategy is to train the models from scratch using only remote sensing data. The second method uses some layers in the pre-trained CaffeNet and GoogLeNet (with ImageNet). The third method completely uses the image features generated from the pre-trained models and only trains the final classifier. The second method has achieved the best performance.

Sometimes transfer learning is not the solution when the recognition task in the target domain can be very different from the recognition task of the existing CNN models. Hence, researchers also build and train new CNN models for their specific learning tasks (e.g., [MMH14, RGC16]. Maire et al. [MMH14] built a CNN to detect marine species in aerial imagery, which outperformed the traditional classifier using features defined by users (e.g., color features, entropy). Romero et al. [RGC16] proposed an unsupervised method using CNN models to extract deep features for remote sensing image classification using a small amount of labeled data with high accuracy (84.53%).

The existing work described here demonstrates the importance of transfer learning and the choice of a training strategy concerning the similarity of the

source and target data domain and the availability of training data in both domains. Section 4.4 will discuss a number of considerations when using pre-trained CNN models and applying transfer learning for map processing given the availability of training data.

4.3 Overview of Semantic Segmentation for Geographic Feature Recognition from Map Scans

The goal of a deep-learning semantic segmentation model (hereafter, semantic segmentation model) is to predict the class label (e.g., people, cars, or buildings) of each pixel in an input image.⁵ Together, the pixels of the same class label constitute one or more detected areas of a specific object class (e.g., red pixels in Fig. 4.1b represent chairs in the image in Fig. 4.1a). In many cases, deep learning models for semantic segmentation build on an existing CNN or use an existing CNN as the building blocks (backbone) in their network architecture (e.g., see [Gar+17, LDY18, KU19]). These existing CNNs are originally designed and pre-trained for image classification tasks (i.e., assigning a class label, e.g., dog, to the entire image), and hence some modifications are required to enable them to handle semantic segmentation tasks.

In general, semantic segmentation models include two main learning objectives. One objective is to determine “what” objects are there in the images (e.g., chair or table objects), which needs semantic information. The other task is to detect “where” are the objects in the images, which needs spatial context (i.e., spatial relationships of object pixels). Note that typically a semantic segmentation model learns to determine “what” and detect “where” simultaneously during the training process. The inference of “what” needs the global information of representative descriptions of image features, which often exist in the deep layers in the neural networks (i.e., the layers after down-sampling). In contrast, the inference of “where” needs fine-grained local information derived from image pixels in a neighborhood, which can be obtained from the shallow layers of fine spatial resolutions in the neural networks. In recent efforts, researchers have been studying how to aggregate the information of “what” and “where” for semantic segmentation (see [KU19, LDY18]).

This section first introduces three widely available, state-of-the-art CNN models for image classification, namely VGG16 [SZ15], GoogLeNet [Sze+15], and ResNet [He+15]. Because of the popularity of these models, their pre-trained versions (usually trained with millions of images from ImageNet [Rus+15b] for image classification) are frequent choices as the network building blocks for deep learning models for many types of image understanding tasks. As an example, Sects. 4.3.1–4.3.3 also include a naive modification to these models for semantic segmentation.

⁵The reader is referred to [Gar+17] for a comprehensive review on semantic segmentation models.

Next, Sects. 4.3.4 and 4.3.5 discuss the two major types of network architectures of deep-learning semantic segmentation models aiming for fusing the information of “what” and “where.” Section 4.3.4 presents the semantic segmentation models that utilize the encoder and decoder structure to recover lost information during the down-sampling process using an example of the Fully Convolutional Networks (FCN) [LSD15]. Section 4.3.4 also includes a more sophisticated modification to VGG16 [SZ15], GoogLeNet [Sze+15], and ResNet [He+15] for semantic segmentation by using a common strategy for fusing the information of “what” and “where.” Section 4.3.5 presents the semantic segment models that use feature images of multi-scale pyramids to preserve both the semantic and spatial information at the same time using the Pyramid Scene Parsing network (PSPNet) as an example [Zha+17]. Section 4.3.5 also includes a modification to PSPNet for geographic feature recognition from historical map scans.

4.3.1 VGG16: The 16-layer Very Deep Convolutional Networks for Large-Scale Image Recognition

Simonyan and Zisserman [SZ15] developed the Very Deep Convolutional Networks (commonly referred to as VGG) that won the first and the second places on two of the image localization and classification tasks in the ImageNet Large-scale Visual Recognition Challenge in 2014 (ILSVRC 2014). The novel contribution of VGG is that it increases the depth of the network (i.e., more layers than exiting CNNs at the time) to improve the image classification performance significantly, while does not increase substantially the computation complexity (i.e., the number of learnable weights in the network) by using very small convolution and pooling filters (3×3 and 2×2 , respectively).

VGG significantly outperformed the previous generation of image classification models that were the winner of the ILSVRC in 2012 and 2013. VGG achieved a 6.8% image classification error in contrast to an 11.7% error from the ILSVRC2013 winner [ZF14, Ser+14]. Figure 4.6 shows a “block” in the VGG network architecture. A “conv” (convolutional) layer (or a filter) contains a set of learnable parameters (or weights). The size of the parameter set depends on the dimension and depth of the layer (e.g., the filter dimension is 3×3 in the case of Fig. 4.6).

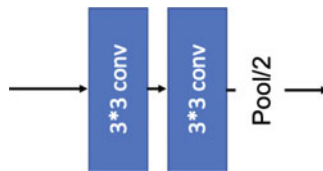


Fig. 4.6 An example block of VGG16: The dimension of filters in a layer is indicated by the number before “conv.” “Pool/2” refers to the stride size of 2 for each max-pooling layer

The “pool” (pooling) layer is commonly used in CNNs between the convolutional layers or blocks to reduce the number of parameters by spatially combining nearby parameters (and hence reduce the spatial resolution). The dimension of the filter, the type of the filter, and the number of layers in the network determine the size of the “receptive field” of the network. For example, each of the output values in the feature image of a 3×3 convolutional layer summarizes a 3×3 region of its input. In the case of VGG, for the first convolutional layer that the input is the raw image, each of the output values in the feature image can be thought of as the image feature learned from multiple 3×3 regions of the input image. This learned image feature is the “local information” (the spatial information of “where”). After multiple layers of convolutional and pooling layers, each of the output values in the feature image represents image features learned from a larger region, and hence the deep layers learn the “global information” (the context information of “what”).

The VGG stacks multiple similar small blocks and connects them with two fully connected (FC) layers (sometimes called dense layers) at the end of the network. The FC layers are commonly used in a neural network to act as the final classifier for generating the classification output. In the case of VGG, the final output is a class label for the entire input image (e.g., an image of the dog class). Similar to typical CNNs, the blocks of convolutional and pooling can be thought of as the extractors or learner of image features, and the FC layers then look at the extracted image features to make a classification decision.

For semantic segmentation tasks, such as extracting pixels of geographic features from historical maps, the goal is to assign a class label to each pixel (image segmentation) instead of one class label for the entire image (image classification). Therefore, a straightforward approach for enabling VGG for semantic segmentation is to first enlarge the output of image features from the network blocks (of convolutional and pooling layers) to the same dimension of the input image with a depth of the number of classes [LSD15]. This way, each pixel in the input image has a set of image features for predicting its class label. This enlargement process can be achieved by simple up-sampling layers (e.g., performing a bilinear interpolation) or deconvolutional layers (with learnable weights as in the convolutional layers). Next, instead of using the FC layers for generating one output for the entire image, a softmax layer takes the output from the enlargement process and generates a probability distribution for each pixel over all possible classes. The summation of the probability of a pixel being in each class is 100%. For example, it could be assumed that the input image has a dimension of 256×256 pixels, and the semantic segmentation model aims to predict three classes: “road,” “text,” and “background.” The output of the enlargement process will have a dimension of 256×256 pixels with a channel depth of 3, each representing the learned image feature of a class. The softmax layer takes this output and again generates a result of a dimension of 256×256 pixels with a channel depth of 3, each representing the probability of a pixel belonging to a specific class. For each pixel in the softmax output, the summation of the class probabilities (along the depth axis) is 100% (e.g., road: 96.1%, text: 2.8%, and background: 1.1%), and the class that has the highest



Fig. 4.7 The modified architecture of VGG16 for semantic segmentation. The number after “conv” represents the number of filters (e.g., “3 × 3 conv, 64” represents 64 convolutional filters of a 3 × 3 dimension)

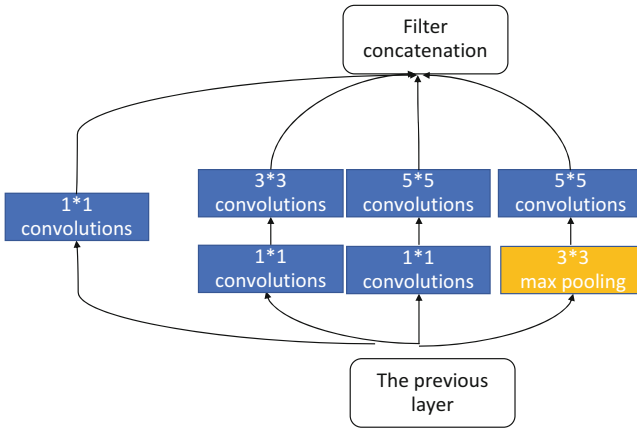


Fig. 4.8 An example of the inception module in GoogLeNet. The numbers represent the filter sizes in the convolutional layers

probability is the predicted class for a pixel. Figure 4.7 shows the architecture of the modified VGG16 for semantic segmentation.

4.3.2 GoogLeNet

Szegedy et al. [Sze+15] proposed the inception module that both increases the depth and the width of a CNN. Figure 4.8 shows the inception module. GoogLeNet is an implementation of the inception module. GoogLeNet won the first place of on one of the classification and localization tasks a ILSVRC in 2014 with a 6.67% top-5 error outperforming VGG16 (a 7.32% top-5 error). GoogLeNet contains 22 layers connecting 9 inception modules after the first few convolutional layers. An inception module contains a number of convolutional filters of different sizes and a concatenation operator that combines the output of the convolutional filters. The idea of the inception module is that instead of using a fixed size convolutional filter, the network can learn to select and combine output from filters of varying sizes. This allows the network to select the most effective sizes of receptive fields and combine the image features learned from these receptive fields. In addition to the inception module, another major difference between GoogLeNet and previous

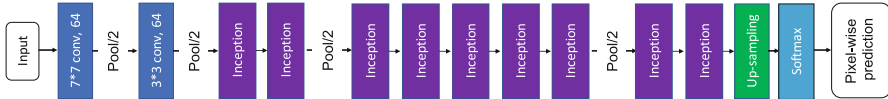


Fig. 4.9 The modified architecture of GoogLeNet for semantic segmentation. The first two convolutional layers have $64 \times 7 \times 7$ filters (“7*7 conv, 64”)

CNNs is that GoogLeNet does not have FC layers. GoogLeNet employs a linear layer of average pooling for the final image classification output. The authors found that the FC layers do not improve the performance but largely increase the number of required parameters (i.e., increase the computational cost). Similar to VGG, for semantic segmentation tasks, the final average pooling layer in GoogLeNet can be replaced with an up-sampling layer to generate an output in the same dimension as the input image with a softmax layer at the end of the network for multi-class classification. Figure 4.9 shows the architecture of the modified GoogLeNet for semantic segmentation.

4.3.3 ResNet

After the success of VGG16 with using deep networks in a CNN, continued experimentation with even deeper networks revealed the problem of degradation, which means that with increasing depth of the network, the accuracy eventually saturates and then degrades rapidly [He+16]. In 2016, He et al. [He+16] presented a landmark paper that applied the residual learning framework, ResNet, to prevent the degradation problem using “shortcut connections” and improve the accuracy of the network without increasing the computation complexity. The shortcut connections enable a residual block to learn the differences between the input and output of the block instead of learning the directing mapping. For example, assuming a network block learns a function $H(x) = y$ where x is the input, and y is the output, in the case of deep networks, the function $H(x)$ is learned from multiple non-linear layers. When the number of layers increases, the network accuracy would improve (comparing to the networks with fewer layers) only to a point and then stop to improve or become worse than that of a shallower network. This is because the direct mapping, $H(x)$, is difficult to learn with a deep network. The shortcut connections solve this problem by allowing the layers in a deep network to use the input from previous layers, and instead of computing $H(x)$ directly, it computes $F(x)$ where $F(x) + x = H(x)$. The function, $F(x)$, is called the residual function and is easier to learn than the direct mapping function $H(x)$. Figure 4.10 shows a block of residual learning in ResNet. ResNet stacks multiple blocks of residual learning and connects them with a final FC layer. ResNet is the winner for image classification task in ILSVPC in 2015 with a 3.57% top-5 error.

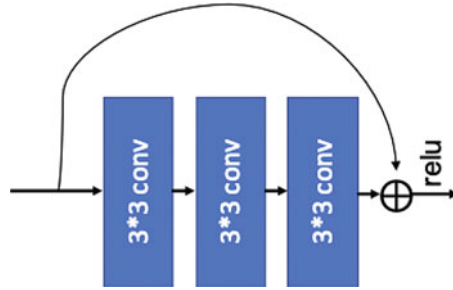


Fig. 4.10 An example block of ResNet. The numbers represent the filter size in the convolutional layers. ReLU (Rectified Linear Units) is the activation function used in ResNet

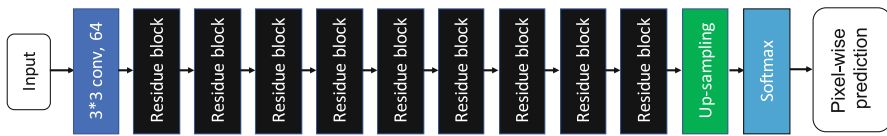


Fig. 4.11 The modified architecture of ResNet for semantic segmentation. The first convolutional layer has $64 \times 3 \times 3$ filters (3×3 conv, 64).

For semantic segmentation tasks, the FC layer in ResNet can be replaced with an up-sampling layer to generate an output in the same dimension as the input image with a softmax layer at the end of the network for multi-class classification. Figure 4.11 shows the architecture of the modified ResNet for semantic segmentation.

4.3.4 The Encoder and Decoder Architecture for Semantic Segmentation

To fuse the semantic (global) and spatial (local) information and recover the loss of spatial information in the deeper layers, one type of semantic segmentation model employs the encoder and decoder structure in a network (e.g., [RFB15, BKC17, CC17, BP18]). The encoder learns the semantic information by gradually reducing the spatial resolution. The corresponding decoder gradually recovers the loss of spatial resolution. For example, SegNet [BKC17] uses a symmetric encoder and decoder architecture and pooling indices in the max-pooling operators to track the spatial locations of the distinctive representations of image features during the learning process. UNet [RFB15] also employs a symmetric architecture, which preserves the local information by concatenating the individual encoder outputs of a higher resolution (i.e., spatial information) to the corresponding decoder output (i.e., semantic information). However, the training process of such symmetric architecture can be inefficient.

Another popular semantic segmentation model with the encoder and decoder structure is the Fully Convolutional Networks (FCN) [LSD15], which use the skip architecture to “skip” layers in the network and directly add outputs from the shallow layers to the deeper layers in the network to overcome the problem of inaccurate segmentation boundaries. The skip architecture combines the final result with the results from the intermediate layers, which preserve some of the location information. Intuitively, combining the results from the intermediate layers and the final layer has the advantage of utilizing both the global and local information since the final convolutional layer represents the global information while the intermediate convolutional layers provide the local information. The experiment results in [LSD15] also show that better segmentation performance can be achieved when more intermediate results are combined with the final results. In contrast, as discussed in the previous sections, VGG (Sect. 4.3.1), GoogLeNet (Sect. 4.3.2), and ResNet (Sect. 4.3.3) are designed for image classification tasks, which require learning the unique image features to distinguish between individual class labels (e.g., dog vs. cat) for the entire input image. Therefore, the loss of local information caused by down-sampling layers (e.g., max-pooling) does not have a significant impact on the classification results. However, image segmentation tasks need to classify each pixel in the input image, which requires local (image) information to precisely delineate the boundaries of the objects. Therefore, the simple modifications of VGG, GoogLeNet, and ResNet (Figs. 4.6, 4.9, and 4.11) only rely on the global information for semantic segmentation and will not be able to generate accurate object boundaries in the semantic segmentation result.

The encoder in FCN can be any CNN, such as VGG, GoogLeNet, or ResNet. In the decoding processing, FCN combines outputs from multiple layers to obtain multi-scale context information, which is called the skip architecture. Figure 4.12 shows the architecture of FCN where the red arrows indicate the skip architecture combining the outputs from multiple layers. The outputs from the shallow layers (e.g., the purple and blue layers) contain fine spatial information because the spatial

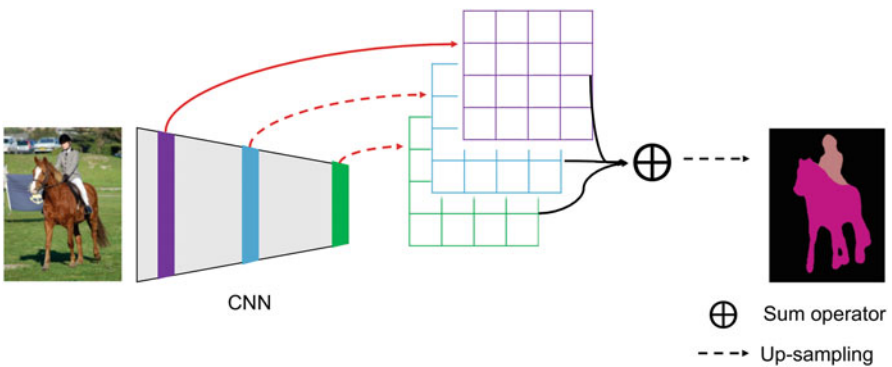


Fig. 4.12 An example of the FCN architecture showing examples of shallow to deep layers (purple, blue, and green layers)

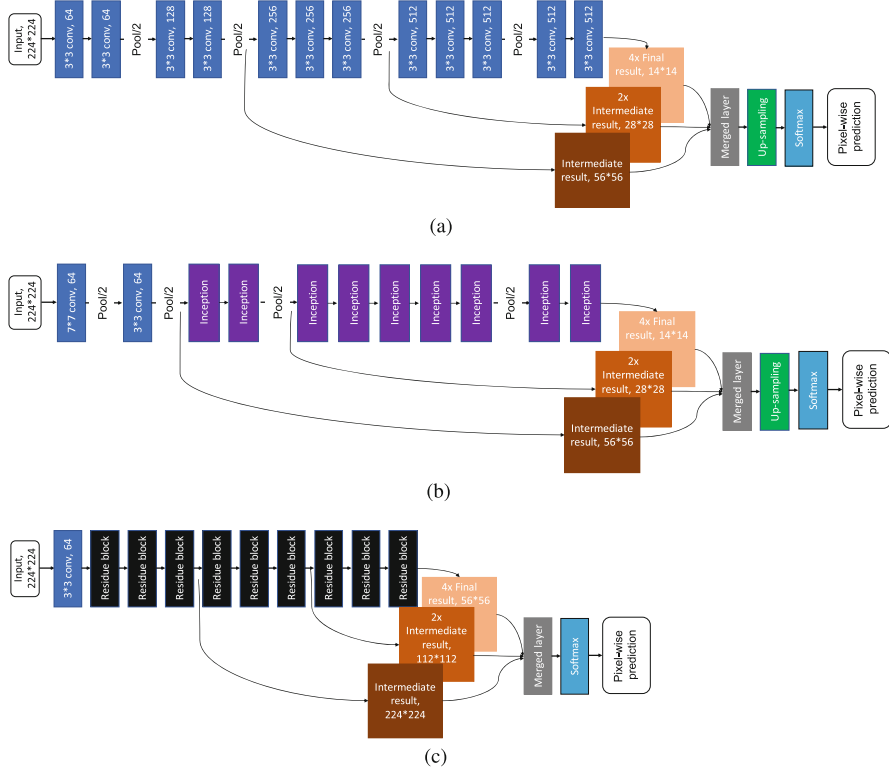


Fig. 4.13 Examples of the modified VGG16, GoogLeNet, and ResNet with the skip architecture for semantic segmentation. Assuming the size of the input image is 224×224 pixels, the networks show the dimensions of the extracted intermediate and final image features for semantic segmentation. For example, dimensions of the intermediate and final image features for the FCN-ResNet are 224×224 , 112×112 , and 56×56 , respectively. (a) FCN-VGG16 with the skip architecture. (b) FCN-GoogLeNet with the skip architecture. (c) FCN-ResNet with the skip architecture

resolution of the outputs is similar to the input images. (Note that the purple layer has the same dimension as the input image, so it does not require up-sampling.) The outputs from the deep layers (e.g., the green layer) have been processed through multiple convolutional layers and max-pooling layers and contain the semantic knowledge for representing the input image but with coarse spatial resolution. Combining the outputs from both shallow and deep layers enables both spatial and semantic information to be used in the final segmentation layer. Figure 4.13 shows the modified FCN-VGG16, FCN-GoogLeNet, and FCN-ResNet with the skip architecture from Figs. 4.7, 4.9, and 4.11.

The main limitation of FCN or FCN-based models is their capability to effectively use the context information in a complex image [Zha+17]. For example, a black line in a map scan following the directions of contour lines is more likely

to represent roads than railroads. Also, the pre-trained CNNs used in FCN usually are trained with lower resolution images while document scans are generally high resolutions images, and hence the pre-trained CNNs might not be able to capture small object details without network modifications.

4.3.5 Multi-Scale Pyramids of Feature Images for Semantic Segmentation

In addition to using an encoder and decoder structure, a type of semantic segmentation architecture utilizes multi-scale pyramids of feature images to capture the spatial and semantic information in the output of each layer in the network [He+16, Yan+18, DDC98]. For example, SPP-Net [He+16] uses spatial pyramid pooling, which employs varying stride sizes to generate image features of multi-scales and then concatenates them into a fixed-length representation. The atrous spatial pyramid pooling [Che+17] applies the atrous convolution operator with varying rates to the same feature map and concatenates them into a new feature map containing both spatial and semantic information. The advantage of spatial pyramid pooling is that it contains semantic information representation of the entire input images instead of only sub-regions determined by the receptive fields as in the case of FCN.

The Pyramid Scene Parsing network (PSPNet) [Zha+17] is a representative model for spatial pyramid pooling and is well suitable to handle complex scenes and a wide range of input data. Figure 4.14 shows the architecture of PSPNet. The CNN component in Fig. 4.14 can be any CNN. The component in the red box in Fig. 4.14 represents the spatial pyramid pooling module. PSPNet uses various pooling layers

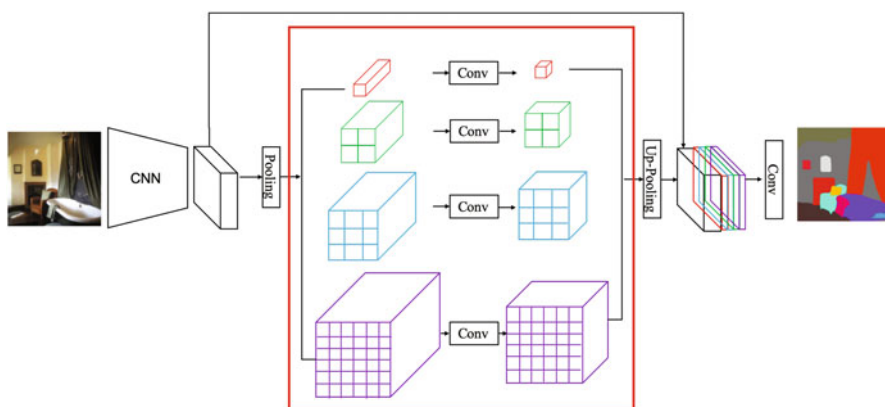


Fig. 4.14 An example of the PSPNet architecture where the red box indicates the spatial pyramid pooling module

to generate feature images with varying resolutions. The dimensions of the feature images from top to bottom in the red box in Fig. 4.14 are 1×1 , 2×2 , 3×3 , and 6×6 . PSPNet uses global average pooling to generate the 1×1 resolution feature image. The 1×1 resolution feature image captures image features representing the entire input images. In contrast, the output of FCN from the deep layers only cover sub-regions of the input images. The image features of the entire input images can help provide global information for classifying pixels that have a similar local context but belong to different object types (e.g., a segment of waterlines vs. characters). After the spatial pyramid pooling module, PSPNet concatenates the feature images generated from CNN and the outputs generated from the spatial pyramid pooling module.

For geographic feature recognition from map scans, the semantic segmentation model needs to be able to delineate small objects from a complex background. This means the learned image features in the network should preserve more spatial information than context information, especially in the case where the number of classes is small. A simple way to preserve the spatial information of the learned image features is to add the skip architecture in the CNN (e.g., a pre-trained VGG used in PSPNet) so that local information can be used in the spatial pyramid pooling module. Figure 4.15 shows the architecture of this modified PSPNet. The trapezium box can be any CNN models, such as VGG. The skip connections combine results from intermediate (green) and final layers to keep both local and global information from the CNN model. The outputs of skip connections, the feature images, are the inputs for the pyramid pooling module.

In sum, the choices of CNNs and the semantic segmentation architecture have a significant impact on geographic feature recognition from historical map scans, especially when the target features are small (e.g., three-pixel wide railroads). Section 4.5 will present a number of experiments for extracting railroads from map tiles of a USGS historical topographic map. The experiments include the FCN models and PSPNet introduced in this section and a summary of their strengths and weaknesses.

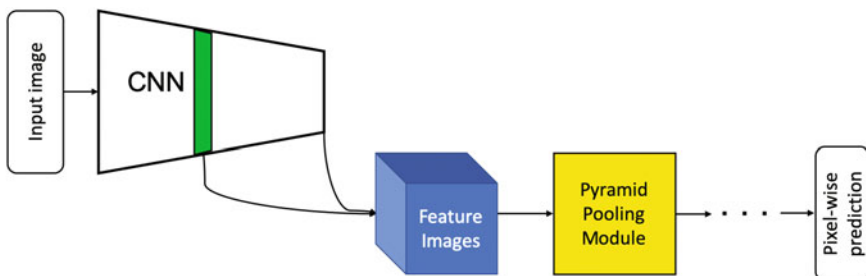


Fig. 4.15 The architecture of the modified PSPNet

4.4 Overview of Transfer Learning for Geographic Feature Recognition from Map Scans

Transfer learning, conceptually, is the process of learning some knowledge from one domain (the source domain) and then applying the learned knowledge to another domain (the target domain). For example, we can train a CNN model to detect if an image contains cats and then fine tune the model for detecting dogs if the training data for cats are abundant and that for dogs are scarce. This section discusses transfer learning strategies concerning the availability of training data and assuming the source and target domains are very different (e.g., images from ImageNet vs. historical maps). The source domain is the domain where a large amount of labeled data is available. The target domain is the domain for which we want to apply the trained networks.

Yosinski et al. [Yos+15] show that the image features learned from the first few layers in a CNN are general features similar to the features learned by classic, curated image recognition features (e.g., Gabor filters or color blobs), while the learned features become more specific to the training data in the deep layers. Hence, the general features learned from the source domain can help the recognition in the target domain even when the target and source domains are very different utilizing a proper strategy of reusing and fine-tuning a pre-trained model [Oqu+14]. For general image understanding applications (e.g., object detection and segmentation), the pre-trained source models are often trained with millions of publicly available image datasets (e.g., ImageNet).

In the case of applying transfer learning to the problem of extracting geographic features from map scans using pre-trained CNNs or semantic segmentation models, the training/testing datasets in the source and target domains can be very different. For example, the source dataset can come from ImageNet, which has over 14 million images and 20,000 classes from daily lives, while the target dataset can be the USGS topographic historical maps. When the source and target datasets are very different, Yosinski et al. [Yos+15] showed that fine-tuning the pre-trained model helps to gain better performance. This process of fine-tuning includes initializing the model parameters using the values learned from ImageNet and using the target dataset to adjust the parameters. However, depending on the size of the training data for the target domain, the fine-tuning process requires careful considerations of a training strategy (e.g., adjusting the weights of shallow vs. deep layers using the training data in the target domain) to obtain the best transfer learning result.

According to the work in [ZF14, Yos+15], the first few layers (the shallow layers) in CNNs learn general features, such as edges. The deeper the layers are, the more specific features the models learn to represent the input images. Hence, when sufficient training data are available for the target domain, fine-tuning a pre-trained model can begin from initializing the model with the pre-trained weights (e.g., pre-trained VGG16 with ImageNet) and then retrain the entire model from the shallow layers to the last layer (classification layer, usually FC and/or softmax layers) using the training data for the target domain. However, if the amount of

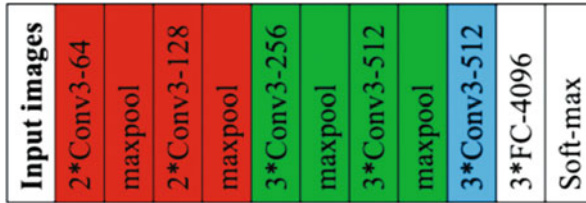


Fig. 4.16 The architecture of VGG16. The label $Z^*ConvX-Y$ means that there are Z convolutional layers in the block, and each convolutional layer has a number of Y filters with the shape $X*X$. The label Z^*FC-X means that there are Z fully connected layers, and each layer has a number of X neurons

the available training data for the target domain is small, training from the shallow layers could result in model overfitting and hence only fine-tuning the deep layer or the classification layer with the training data for the target domain is preferred. As an example, Fig. 4.16 shows the architecture of VGG with examples of color-coded shallow, middle, deep, and classification layers (the red, green, blue, and white boxes, respectively).

The capability to transfer the learned knowledge from a source domain to a target domain is important because when the training data from the target domain is scarce, the computation resource is limited, or the training time is constrained, learning the “knowledge” from the target domain directly is impractical. For example, VGG16 has 16.4 million parameters, which is not feasible to learn directly from only thousands of training examples. With many pre-trained CNNs available online (e.g., VGG, GoogLeNet, and ResNet), reusing these pre-trained CNNs for digital map processing not only saves resources in creating large numbers of map training data but also prevents the need for expensive computing resources for training a CNN from scratch. Section 4.5 will present a number of experiments for geographic feature recognition from map scans. The experiments vary the size of the available training data and test transfer learning strategies of fine-tuning the shallow, middle, deep, and classification layers.

4.5 Experiment

This section presents three sets of experiments. In the context of railroad recognition from historical map scans, the experiments test the impact of (1) different types of CNN backbones in semantic segmentation models, (2) various transfer learning strategies, and (3) modifications on PSPNet for improved feature recognition from historical map scans.

4.5.1 Experimental Data, Settings and Evaluation Metrics

The experimental data contain eight sections of map tiles cropped from a scanned USGS historical topographic map in the 7.5-min series (scale 1:24,000) downloaded from the USGS website (Fig. 4.17). The scan resolution is around 600 DPI (dots-per-inch). The image dimension of the entire map scan (including the map collar) is $12,943 \times 16,188$ pixels. The map title is Bray Quadrangle, which covers Bray, California, USA (originally produced by the USGS in 1988 with revision by USDA Forest Service in 2001). Each of the eight sections of the test map tiles has a similar image dimension. Together the eight map tiles constitute the entire map scan. All the deep learning models in the experiments in this section were trained on seven map tiles and tested on the one remaining tile.

The test map (Bray Quadrangle) contains rich and diverse graphical compositions and geographic features. For example, the test map contains geographic features of railroads, elevation contours, rivers, roads, trails, highways, and buildings. Figure 4.18 shows examples of geographic features in the test map. In addition, the graphical compositions surrounding railroads in the test map are also diverse. For example, in Fig. 4.19, there are elevation contour lines, highways, roads, and hydrographic features around railroads on the test map. These complex compositions of cartographic symbols make the test map an ideal test case of using a deep learning semantic segmentation model for feature recognition from historical map scans.

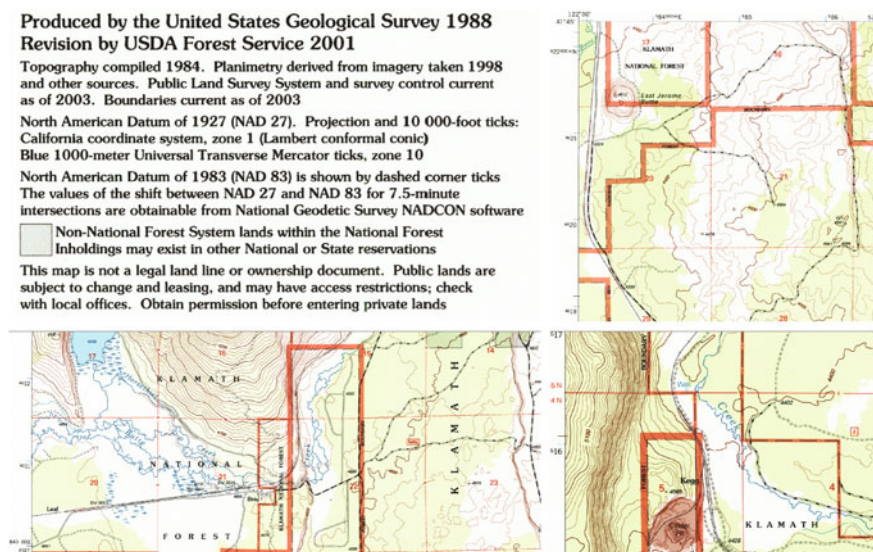


Fig. 4.17 The metadata and example sections of the test map (USGS historical topographic map, Bray, California (circa 2001))

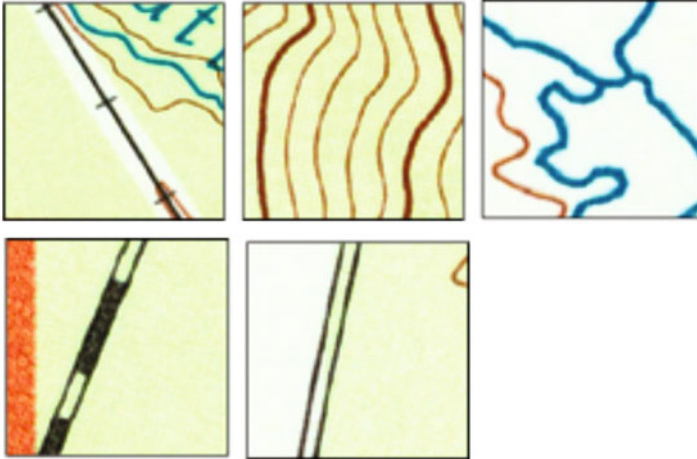


Fig. 4.18 Examples of geographic features in the test map: railroads, elevation contour lines, hydrography, highways, and roads (from left to right and top to bottom)



Fig. 4.19 Examples of the diverse background surrounding railroads (black lines with crosses) in the test map

Training, validating, and testing semantic segmentation models require ground truth data indicating the class label of each pixel in the test map (railroads vs. non-railroads). The ground truth data were generated from manually created one-pixel-wide railroad centerlines for the entire test map. After tracing the centerlines, the one-pixel lines were then buffered using one pixel on each side to create three-pixel-wide linear areas in an annotation image to represent the ground truth. The annotation image has the same image dimension as the test map. In the annotation image, the pixels in the buffered three-pixel-wide areas represent the approximate locations of railroad symbols in the map, and other pixels represent non-railroad content.⁶

⁶Note that using the centerlines with a buffer helps generate approximated pixel-wise annotations of the railroad symbols efficiently, and manually annotating each railroad pixels in the map is impractical even for one map scan.

For training the semantic segmentation models in this section, the training data were generated using the seven training map tiles and the annotation image. The training map tiles and their corresponding areas in the annotation image were cropped into small images of 320×320 pixels for batch training. The annotation image provides a class label (railroads vs. non-railroads) for each pixel in the training map tiles. The map pixel having its corresponding pixel in the buffered railroad centerlines in the annotation image is a railroad pixel; otherwise, it is a non-railroad pixel. For example, a map pixel at the location $(x = 2, y = 3)$ (in the image coordinates) is a railroad pixel if the pixel at the location $(x = 2, y = 3)$ in the annotation image belongs to the buffered railroad centerlines. In addition, the cropped images were rotated four times with four angles randomly selected from 0 to 180° for training data augmentation to simulate railroads of various rotation angles in the map.

In the testing phase, the test map tile was cropped using a sliding window with a stride of 50 pixels. The trained segmentation models were then used to classify each pixel in the cropped windows to determine railroad pixels.

All three sets of experiments use the conventional semantic segmentation metric, Intersection over Union (IoU), to evaluate the extracted railroad layer from the test map tile. IoU differentiates three types of pixels in the extraction results. The first type is the correctly classified pixels: the true positives (TP). The second type is the pixels that belong to railroads but misclassified as non-railroads: the false negatives (FN). The third type is the pixels that belong to non-railroads but misclassified as railroads: the false positives (FP). IoU is calculated as the number of TP divided by the sum of the numbers of TP, FP, and FN.

4.5.2 Experiment I: The Impact of Backbone CNNs: FCN-VGG16, FCN-GoogLeNet, and FCN-ResNet

To compare the impact of the architectures of CNN backbones in semantic segmentation models for geographic feature recognition, three FCN variations were tested, including FCN-VGG16, FCN-GoogLeNet, and FCN-ResNet (Fig. 4.13), all with pre-trained weights from ImageNet using the pre-trained models from Keras.⁷ In the training phase, a number of 4800 training samples were randomly selected from all training map tiles. The FCN models were initialized with their pre-trained weights and then trained from the shallow layers using the map training data.

Table 4.1 shows the resulting IoU for the three FCN models on the test map tile. Figure 4.20⁸ provides the visualization of the test map with segmentation

⁷Keras is a high level deep learning Python library (<https://keras.io>) Currently, Keras provides a number of pre-trained CNN models (using ImageNet) for image classification, including VGG16, VGG19, ResNet, ResNetV2, ResNeXt, GoogLeNet, MobileNet, DenseNet, and NASNet.

⁸Figures A.1, A.2, and A.3 in the appendix show the full size results.

Table 4.1 Semantic segmentation results of FCN-VGG16, FCN-GoogLeNet, and FCN-ResNet on the test map tile

	FCN-VGG16	FCN-GoogLeNet	FCN-ResNet
IoU	4.689%	11.79%	23.09%

results where red pixels represent false positives, and the blue pixels represent true positives, and the black lines with crosses are railroads.⁹ The segmentation results from the three models are consistent with their previous image classification performance on ImageNet. The IoU of FCN-ResNet is over 100% higher than that of FCN-GoogLeNet and over 500% higher than that of FCN-VGG. Figure 4.20 shows that FCN-ResNet recognized railroads with diverse background (white, pink, and green) compared to FCN-GoogLeNet and FCN-VGG (only detected a very small portion of the railroads, e.g., small TP areas pointed by the white arrows). However, the FCN-ResNet still only achieved a 23.09% IoU, which could be a result of the lack of training data and the models' limited capability for delineating accurate boundaries and detecting small objects. For example, Fig. 4.20 shows that very often the width of recognized railroads (the red pixels and the blue pixels) is much wider than the width of railroads on the map (i.e., many red pixels, FP, around railroad lines).

In sum, this first set of experiment shows that the choice of the CNN backbone can have a significant impact on the segmentation performance, and the CNN backbones with low error rates on image classification is a better option for semantic segmentation on historical maps than the ones with a higher error rate.

4.5.3 Experiment II: The Impact of Transfer Learning Strategies: PSPNet

The second set of experiments used a pre-trained PSPNet¹⁰ on PASCAL VOC 2012 and then fine-tuned the pre-trained model using various training strategies on map data to demonstrate how transfer learning strategies could affect the recognition results. In the training phase, PSPNet was initialized with the weights from the pre-trained models and then trained from the shallow, middle, deep, and classification layers¹¹ with increasing amounts of training data (small, medium, and large sizes) from historical maps. Table 4.2 shows the number of training samples in the small, medium, and large sizes of training data after data augmentation. The training data

⁹The blue pixels are at the railroad locations in the map since they are correctly extracted railroad pixels.

¹⁰<https://github.com/hszhao/PSPNet/>.

¹¹Figure 4.16 shows the shallow, middle, and deep layers in the backbone CNN of PSPNet. The red, green, and blue boxes represent shallow, middle, and deep layers, respectively.

Fig. 4.20 Semantic segmentation results using FCN-VGG16, FCN-GoogLeNet, and FCN-ResNet (top left, top right, bottom, respectively) (FP in red and TP in blue). The small figures next to each big figures magnify a region of the test map tile

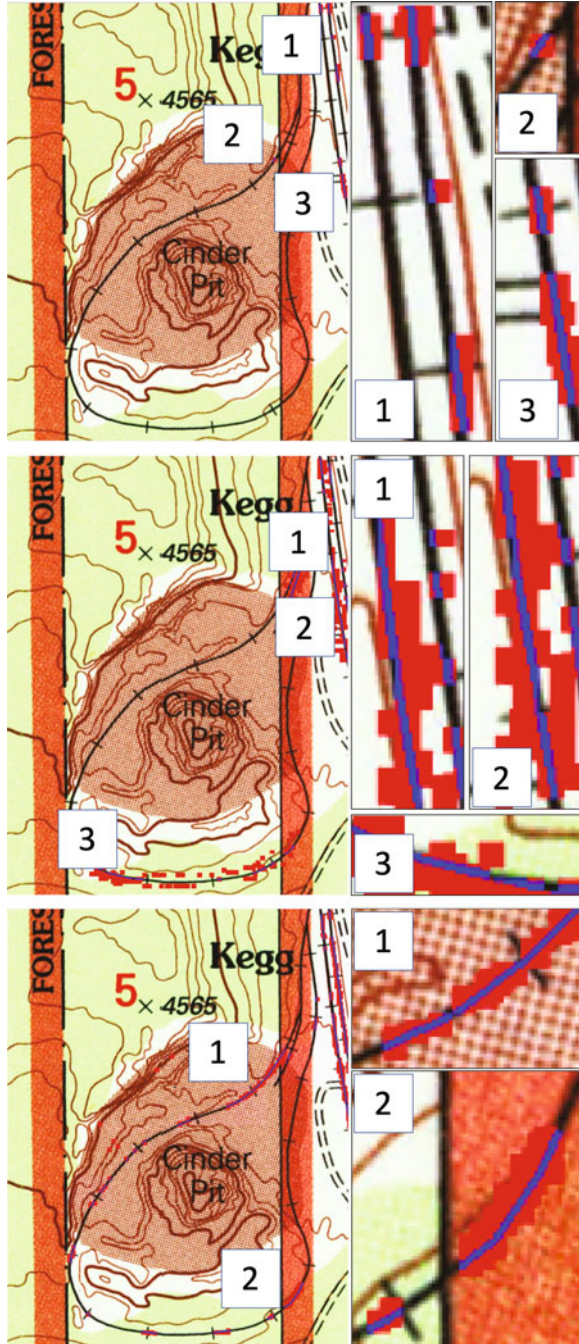


Table 4.2 The number of training samples in different sizes of the training data

	Small	Medium	Large
Number of training samples	1200	2400	4800

Table 4.3 The PSPNet semantic segmentation IoU results

	Classification layer	Deep layer	Middle layer	Shallow layer
Small size training data	0%	0.9337%	1.527%	1.960%
Medium size training data	0%	3.078%	13.09%	26.35%
Large size training data	0%	2.612%	16.42%	29.04%

for each size were randomly selected from the seven training map tiles. The training process used a batch size of 20 training images and stochastic gradient descent as the optimizer with the learning rate as 0.0005. The binary cross-entropy was used as the loss function. The model was trained with 300 epochs until convergence.

Table 4.3 shows the segmentation results in IoU from the various combinations of training data sizes and transfer learning strategies. First, training from the shallow layers outperformed all other training strategies with every training data size. This is because the pre-trained model was trained on very different images from historical map content, and hence training only a few layers (e.g., from the classification and deep layers) with map data would not help recognize railroads from the test map tile. Training from the shallow layers allows more parameters in the network to be adjusted to learn feature representations for the target domain (i.e., railroads on the test map tile).

As for the size of the training data, the IoU performance obtained by training the small size of data was poor because the small size of training data could not cover the diversity of the railroads in the test map tile. Figure 4.21 shows that the segmentation model trained with the small size training data only recognized a few railroad segments to the right of the test map tile. The IoU performance was improved when the number of trainable parameters and the size of training data was increased (Table 4.3 and Figs. 4.21, 4.22, and 4.23).¹² In this experiment, the best IoU was achieved with training from the shallow layers using the medium or large size of training data (26.35% and 29.04%, respectively). However, the best results are similar to that of FCN-ResNet, and all of the tested models and methods so far have the same problem of inaccurate boundaries and wider extraction results than the ground truth (i.e., many false positives).

In sum, because of the significant difference between images in public datasets and historical map scans, only fine-tuning the classification or from deep layers would not work. In addition, selecting the best transfer learning strategy depends on the available size of training data and computing resource for training. Table 4.3 shows that IoU only increased slightly (26.35–29.04%) when using the large size

¹²Figures A.4, A.5, A.6, A.7, A.8, A.9, A.10, A.11, and A.12 in the appendix show the full size results.

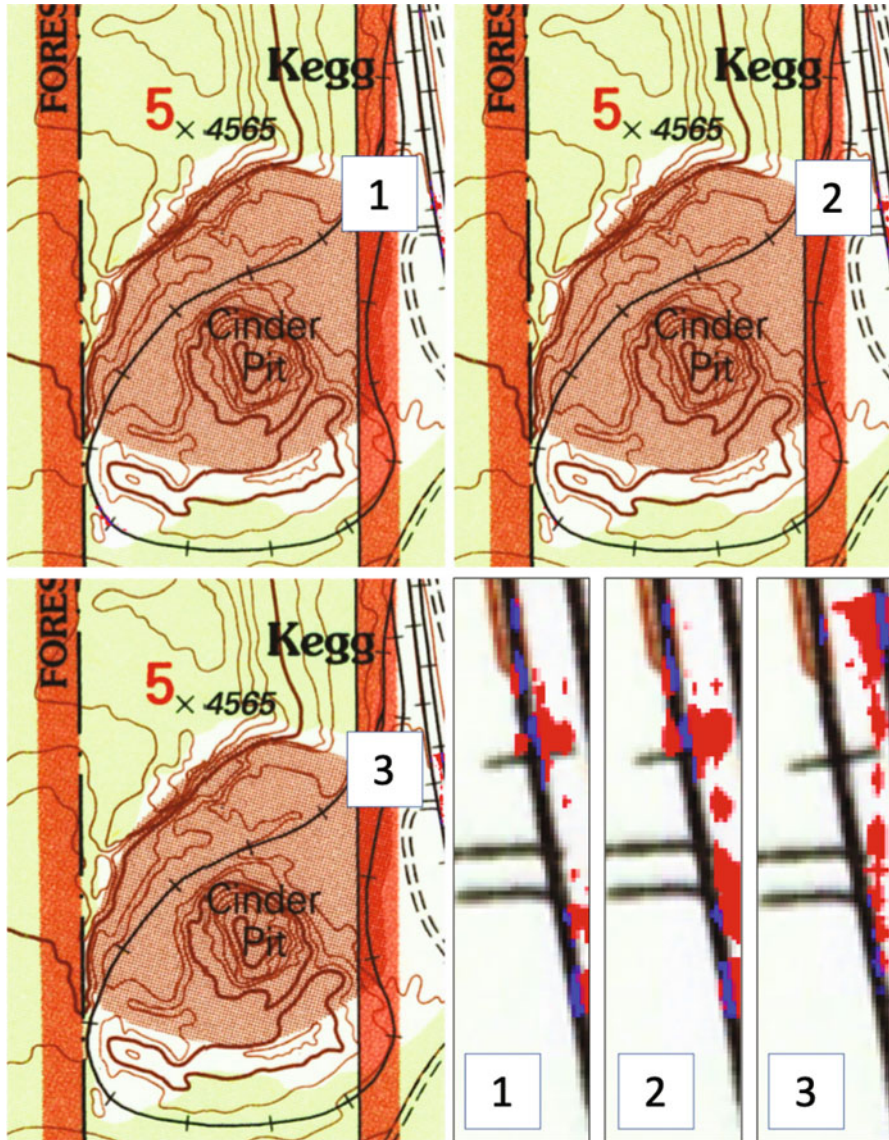


Fig. 4.21 Semantic segmentation results (overlying with the test map) using pre-trained PSPNet with **small size training data** and trained from the shallow (top left), middle (top right), and deep layers (bottom left) (FP in red and TP in blue)

training data instead of the medium size training data while training from the shallow layers. This could mean that the medium size training data could already represent the diversity of the railroads in the test map tile and using the large size training data did not contribute to a significant improvement. As a result, training

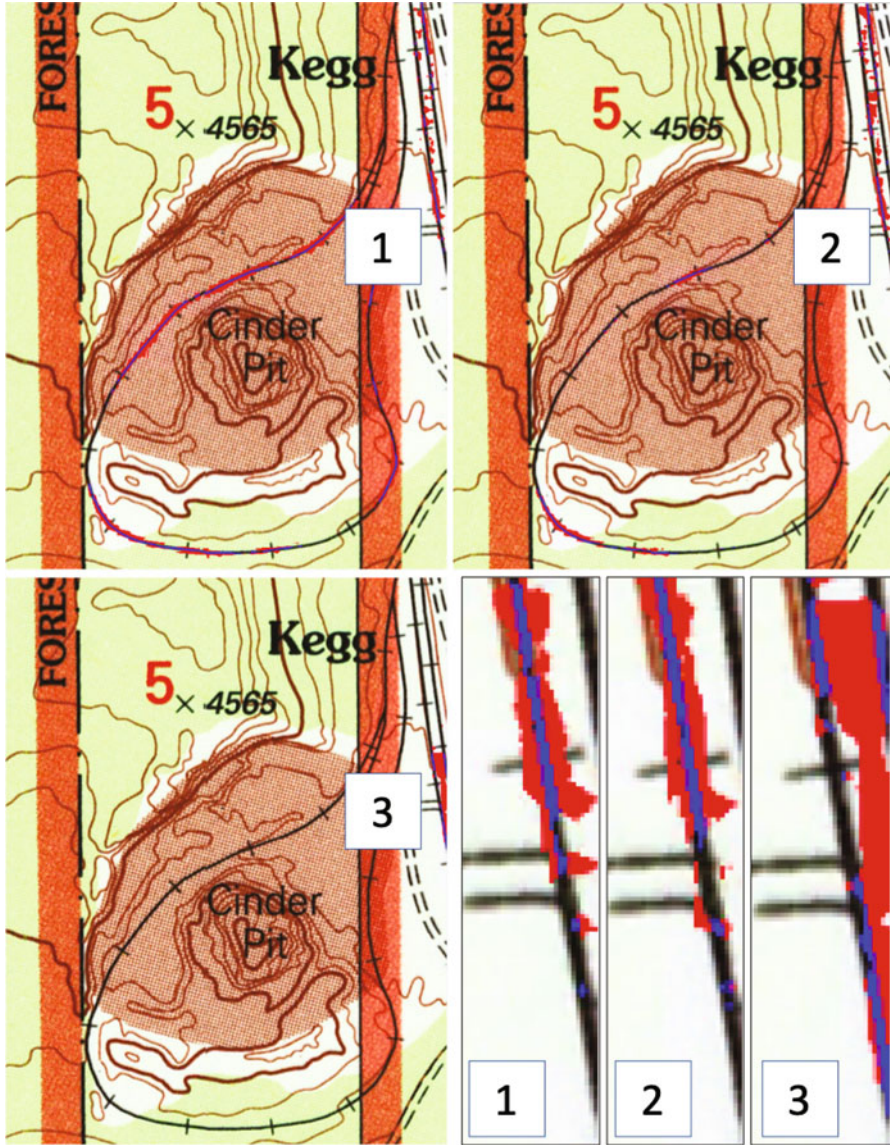


Fig. 4.22 Semantic segmentation results (overlying with the test map) using pre-trained PSPNet with **medium size training data** and trained from the shallow (top left), middle (top right), and deep layers (bottom left) (FP in red and TP in blue)

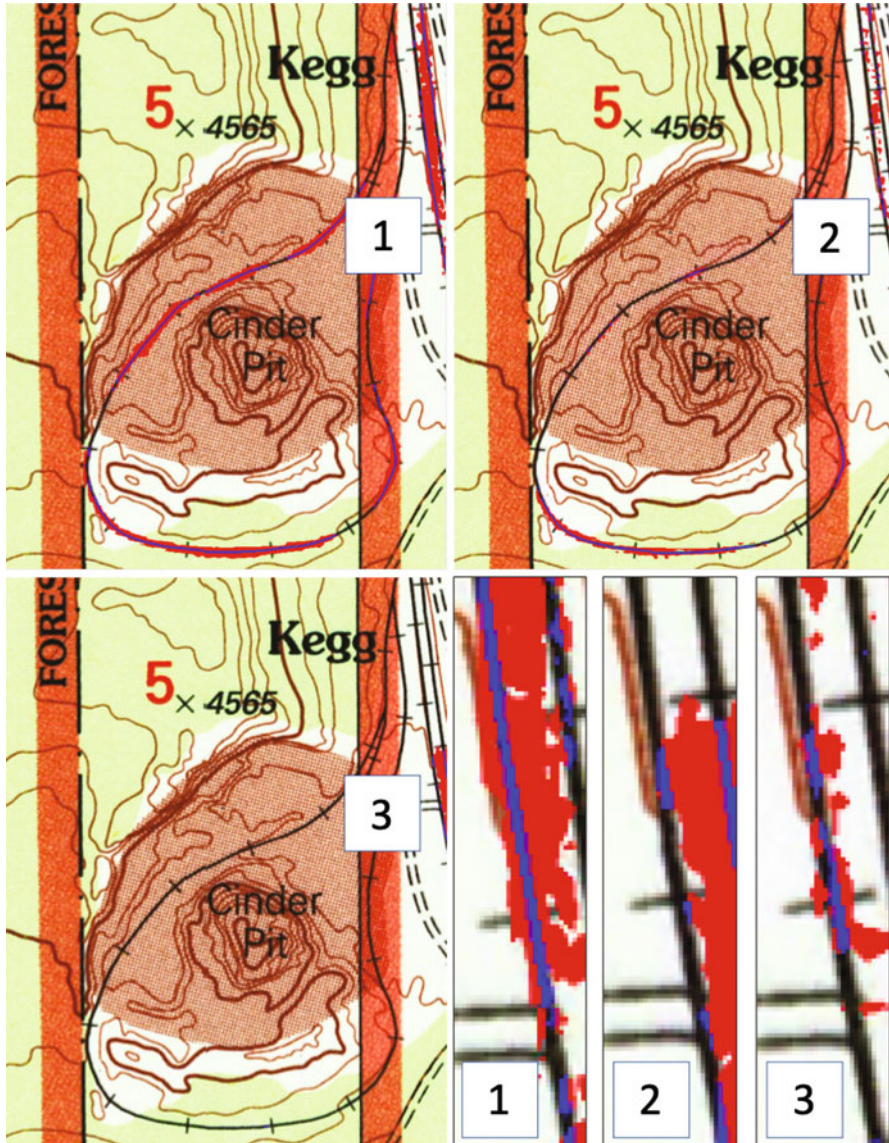


Fig. 4.23 Semantic segmentation results (overlying with the test map) using pre-trained PSPNet with **large size training data** and trained from the shallow (top left), middle (top right), and deep layers (bottom left) (FP in red and TP in blue)

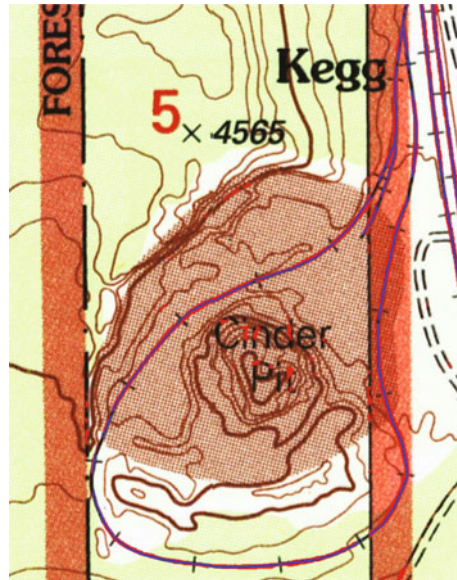
a pre-trained model (from the public datasets) from the shallow layers using the medium size training data can be the most efficient transfer learning strategy with comparable results as using the large size training data. Training from the shallow layers using the large size training data is the best transfer learning strategy if resources are available.

4.5.3.1 Experiment III: Modified PSPNet

Although FCN-ResNet and PSPNet with an appropriate transfer learning strategy could recognize diverse types of railroad pixels on the test map tile in the first and second set of experiment, the delineated boundaries of railroads were still rough with low IoU numbers. This is because these existing segmentation models and the backbone CNNs have many convolutional and pooling layers, which makes it difficult to recover the detailed spatial locations of objects, especially for small objects in maps. The modified PSPNet with skip connections in VGG (Fig. 4.15) combines both local and global information before the spatial pyramid pooling module to overcome this difficulty.

In this experiment, the modified PSPNet was trained with the large size training data for railroads from the shallow layers. The IoU performance of this model was 62.22%, which significantly outperformed other models tested in this section (29.04%). Figure 4.24 shows that, with the modified PSPNet, the recognized boundaries (the red and blue lines, false positives and true positives, respectively) are much closer to the boundaries of railroads than the results from all other models

Fig. 4.24 Semantic segmentation results (overlying with the test map) using the pre-trained, modified PSPNet **large size training data** and **trained from the shallow layers** (red: FP; blue: TP; black lines with crosses: railroads)



tested in this section. However, an IoU of 62.22% still would not be able to generate usable railroad data from historical maps, and more advanced methods are required to extract boundaries of small objects from historical map scans accurately.

4.6 Chapter Summary

This chapter introduced an overview of deep-learning semantic segmentation models and transfer learning strategies and discusses their strengths and weaknesses concerning geographic feature recognition from historical map scans. This chapter also presented a comprehensive experiment on geographic feature recognition from historical maps using railroad features and USGS historical topographic maps as a case study. The experiment showed that for geographic feature recognition from historical maps, fine-tuning the classification or middle layers of pre-trained models alone would not work, no matter the size of the training data from the target domain. The best recognition results (high true positives and low false positives) can be achieved by fine-tuning the entire network with large amounts of training data. These experiments demonstrated that state-of-the-art semantic segmentation models still do not allow to delineate the precise boundaries of geographic features in historical maps, even with some modification to preserve the local information. To fully take advantage of the valuable content in historical map series, advanced semantic segmentation methods that can handle small objects and extract precise boundaries still need to be developed. However, the studies described here provide interesting directions and significantly improved recognition results.

References

- [ALL17] N. Audebert, B. Le Saux, S. Lefèvre, Semantic segmentation of earth observation data using multimodal and multi-scale deep networks, in: *Computer Vision – ACCV 2016*, ed. by S.-H. Lai, V. Lepetit, K. Nishino, Y. Sato (Springer, Cham, 2017), pp. 180–196. ISBN: 978-3-319-54181-5
- [Bal+19] V.E. Balas, S.S. Roy, D. Sharma, P. Samui (eds.), *Handbook of Deep Learning Applications* (Springer, Berlin, 2019). ISBN: 978-3-030-11478-7. <https://doi.org/10.1007/978-3-030-11479-4>
- [BKC17] V. Badrinarayanan, A. Kendall, R. Cipolla, Segnet: a deep convolutional encoder-decoder architecture for image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **39**(12), 2481–2495 (2017)
- [BP18] P. Bilinski, V. Prisacariu, Dense decoder shortcut connections for single-pass semantic segmentation, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2018), pp. 6596–6605
- [Bro+08] G.J. Brostow, J. Shotton, J. Fauqueur, R. Cipolla, Segmentation and recognition using structure from motion point clouds, in *European Conference on Computer Vision* (Springer, Berlin, 2008), pp. 44–57
- [BST15] G. Bertasius, J. Shi, L. Torresani, High-for-low and low-for-high: efficient boundary detection from deep object features and its applications to high-level vision, in

- Proceedings of the IEEE International Conference on Computer Vision* (2015), pp. 504–512
- [Cas+15] M. Castelluccio, G. Poggi, C. Sansone, L. Verdoliva, Land use classification in remote sensing images by convolutional neural networks. arXiv: 1508.00092 (2015)
- [CC17] A. Chaurasia, E. Culurciello, LinkNet: exploiting encoder representations for efficient semantic segmentation, in *2017 IEEE Visual Communications and Image Processing, VCIP 2017* (St. Petersburg, 2017), pp. 1–4
- [Che+17] L. Chen, G. Papandreou, F. Schroff, H. Adam, Rethinking atrous convolution for semantic image segmentation. arXiv: 1706.05587 (2017)
- [CK13] Y.-Y. Chiang, C. A. Knoblock, A general approach for extracting road vector data from raster maps. *Int. J. Doc. Anal. Recognit.* **16**(1), 55–81 (2013). <https://doi.org/10.1007/s10032-011-0177-1>
- [CLK14] Y.-Y. Chiang, S. Leyk, C.A. Knoblock, A survey of digital map processing techniques. *ACM Comput. Surveys* **47**(1), 1–44 (2014). ISSN: 0360-0300. <https://doi.org/10.1145/2557423>
- [DDC98] D. Bin, B. Ding, W.K. Cheong, A system for automatic extraction of road network from maps, in *Proceedings of the IEEE International Joint Symposia on Intelligence and Systems (Cat. No.98EX174)* (1998). <https://doi.org/10.1109/ijsis.1998.685476>
- [Dua+18] W. Duan, Y. Chiang, C.A. Knoblock, S. Leyk, J. Uhl, Automatic generation of precisely delineated geographic features from georeferenced historical maps using deep learning, in *Proceedings of the AutoCarto* (2018)
- [Eve+] M. Everingham, L. Van Gool, C.K.I. Williams, J. Winn, A. Zisserman, *The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results*
- [EW11] M. Everingham, J. Winn, The PASCAL visual object classes challenge 2012 (VOC2012) development kit, in *Pattern Analysis, Statistical Modelling and Computational Learning, Tech. Rep* (2011)
- [Fel98] C. Fellbaum. *WordNet: An Electronic Lexical Database* (Bradford Books, 1998)
- [Gar+17] A. Garcia-Garcia, S. Orts-Escolano, S. Oprea, V. Villena-Martinez, J. Garcia-Rodriguez, A review on deep learning techniques applied to semantic segmentation. arXiv preprint arXiv:1704.06857 (2017)
- [He+15] K. He, X. Zhang, S. Ren, J. Sun, Delving deep into rectifiers: surpassing human-level performance on ImageNet classification, in *Proceedings of the IEEE International Conference on Computer Vision* (2015), pp. 1026–1034
- [He+16] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2016), pp. 770–778
- [Hen+09] T.C. Henderson, T. Linton, S. Potupchik, A. Ostanin, Automatic segmentation of semantic classes in raster map images, in *IAPR International Workshop on Graphics REcognition* (2009), pp. 1–10
- [Hua+17] G. Huang, Z. Liu, L. Van Der Maaten, K.Q. Weinberger, Densely connected convolutional networks, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2017), pp. 4700–4708
- [Jia+17] Y. Jiang, X. Zhu, X. Wang, S. Yang, W. Li, H. Wang, P. Fu, Z. Luo, R2CNN: rotational region CNN for orientation robust scene text detection. arXiv: 1706.09579 (2017)
- [KU19] Ç. Kaymak, A. Uçar, A brief survey and an application of semantic image segmentation for autonomous driving, in *Handbook of Deep Learning Applications* (Springer, Berlin, 2019), pp. 161–200
- [LDY18] X. Liu, Z. Deng, Y. Yang, Recent progress in semantic image segmentation, in *Artificial Intelligence Review* (2018). ISSN: 1573-7462. <https://doi.org/10.1007/s10462-018-9641-3>
- [LSD15] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2015), pp. 3431–3440

- [Mar+16] D. Marmanis, M. Datcu, T. Esch, U. Stilla, Deep learning earth observation classification using ImageNet pretrained networks. *IEEE Geosci. Remote Sens. Lett.* **13**(1), 105–109 (2016)
- [MMH14] F. Maire, L. Mejias, A. Hodgson, A convolutional neural network for automatic analysis of aerial imagery, in *2014 International Conference on Digital Image Computing: Techniques and Applications (DICTA)* (IEEE, Piscataway, 2014), pp. 1–8
- [Oqu+14] M. Oquab, L. Bottou, I. Laptev, J. Sivic, Learning and transferring mid-level image representations using convolutional neural networks, in *2014 IEEE Conference on Computer Vision and Pattern Recognition* (2014), pp. 1717–1724. <https://doi.org/10.1109/CVPR.2014.222>
- [RFB15] O. Ronneberger, P. Fischer, T. Brox, U-net: convolutional networks for biomedical image segmentation, in *International Conference on Medical Image Computing and Computer-Assisted Intervention* (Springer, Berlin, 2015), pp. 234–241
- [RGC16] A. Romero, C. Gatta, G. Camps-Valls, Unsupervised deep feature extraction for remote sensing image classification. *IEEE Trans. Geosci. Remote Sens.* **54**(3), 1349–1362 (2016)
- [Rus+15a] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A.C. Berg, L. Fei-Fei, ImageNet large scale visual recognition challenge. *Int. J. Comput. Vis.* **115**(3), 211–252 (2015). <https://doi.org/10.1007/s11263-015-0816-y>
- [Rus+15b] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al. ImageNet large scale visual recognition challenge. *Int. J. Comput. Vis.* **115**(3), 211–252 (2015)
- [Ser+14] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, Y. Le-Cun, OverFeat: integrated recognition, localization and detection using convolutional networks, in *2nd International Conference on Learning Representations (ICLR 2014). Conference Track Proceedings* (Banff, 2014)
- [SH12] R. Samet, E. Hancer, A new approach to the reconstruction of contour lines extracted from topographic maps. *J. Vis. Commun. Image Represent.* **23**(4), 642–647 (2012)
- [SZ15] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, in *3rd International Conference on Learning Representations, ICLR 2015, Conference Track Proceedings* (San Diego, 2015)
- [Sze+15] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, Going deeper with convolutions, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2015), pp. 1–9
- [Uhl+17] J.H. Uhl, S. Leyk, Y.-Y. Chiang, W. Duan, C.A. Knoblock, Extracting human settlement footprint from historical topographic map series using context-based machine learning, in *IET Conference Proceedings* (2017)
- [Yan+18] X. Yang, H. Sun, K. Fu, J. Yang, X. Sun, M. Yan, Z. Guo, Automatic ship detection in remote sensing images from Google Earth of complex scenes based on multiscale rotation dense feature pyramid networks. *Remote Sens.* **10**(1), 132 (2018)
- [YK16] F. Yu, V. Koltun, Multi-scale context aggregation by dilated convolutions, in *4th International Conference on Learning Representations, ICLR 2016, Conference Track Proceedings* (San Juan, 2016)
- [Yos+15] J. Yosinski, J. Clune, A.M. Nguyen, T.J. Fuchs, H. Lipson, Understanding neural networks through deep visualization. arXiv: 1506.06579 (2015)
- [ZF14] M.D. Zeiler, R. Fergus, Visualizing and understanding convolutional networks, in *European Conference on Computer Vision* (Springer, Berlin, 2014), pp. 818–833
- [Zha+17] H. Zhao, J. Shi, X. Qi, X. Wang, J. Jia, Pyramid scene parsing network, in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition* (2017), pp. 2881–2890
- [Zho+17] X. Zhou, C. Yao, H. Wen, Y. Wang, S. Zhou, W. He, J. Liang, EAST: an efficient and accurate scene text detector, in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition* (2017), pp. 5551–5560

Chapter 5

Summary and Discussion



Abstract This chapter summarizes the book and provides a brief outlook.

5.1 Book Summary

This book presented the challenges and best practices in working with digital historical maps in geospatial environments, aiming to bridge the gap between the user and developer communities of map processing technologies. Historical maps are unique witnesses of the evolution of the past landscape, cultural heritage, and human activities. They are valuable for researchers in a variety of communities ranging from urban geography to landscape ecology. However, extracting information from historical maps to make it available as spatial layers in Geographical Information Systems (GIS) remains a major challenge. This is because of the complexity of the map contents, the low graphical quality of scanned map documents, and the sheer data volume considering that millions of scanned maps have been made publicly available by various mapping agencies and private institutions.

Over several decades, map processing research has focused on technical and methodological solutions to preprocess map scans, extracting map layers efficiently, and georeferencing map sheets. These research advances, at the intersection of computer science and geographic information science, provided the basis for the development of semi- and fully automated map processing solutions with remaining limitations. On the one hand, fully automated map processing solutions often only handle specific types of historical maps and graphical conditions. On the other hand, semi-automated solutions often handle various types of historical maps and graphical conditions but do not scale well for processing large numbers of maps. These persistent limitations impede the extensive use of historical map data in research that involves large geographical extents or long periods of time. Even with recent advances in machine learning for pattern recognition, remote sensing, and computer vision, those limitations remain unsolved, mainly because such techniques do not perform well on processing historical map documents without further adjustment. The main reasons are the general lack of useful training data necessary for successful learning of the graphical characteristics of map layers and

symbols, and the high accuracy requirements in extracting boundaries of highly detailed, irregular, and small geospatial features from maps. This is significant because the user of spatial information expects high accuracy, high precision, and high degrees of completeness, which is different from most other recognitions tasks.

Current efforts show promising results in tackling these issues in map processing by incorporating contextual data, directly or indirectly, as training data and through multi-scale deep-learning architectures for semantic segmentation that show high accuracies and have the potential to preserve object boundaries at a fine spatial resolution. Further advances in map processing and recognition will result in the creation of geospatial data collections representing map contents and firmly connect the user and developer communities thus generating exciting research opportunities enriched by unique historical information from maps. These research opportunities will advance our knowledge of the socio-environmental systems and their changes over long time periods. Thus, future research will be able to evaluate implications and long-term effects of human activities in unprecedented ways by getting access to 200 years' worth of data layers describing the evolution of the landscape.

Appendix A

Railroad Recognition Results

See Figs. [A.1](#), [A.2](#), [A.3](#), [A.4](#), [A.5](#), [A.6](#), [A.7](#), [A.8](#), [A.9](#), [A.10](#), [A.11](#), [A.12](#), and [A.13](#).

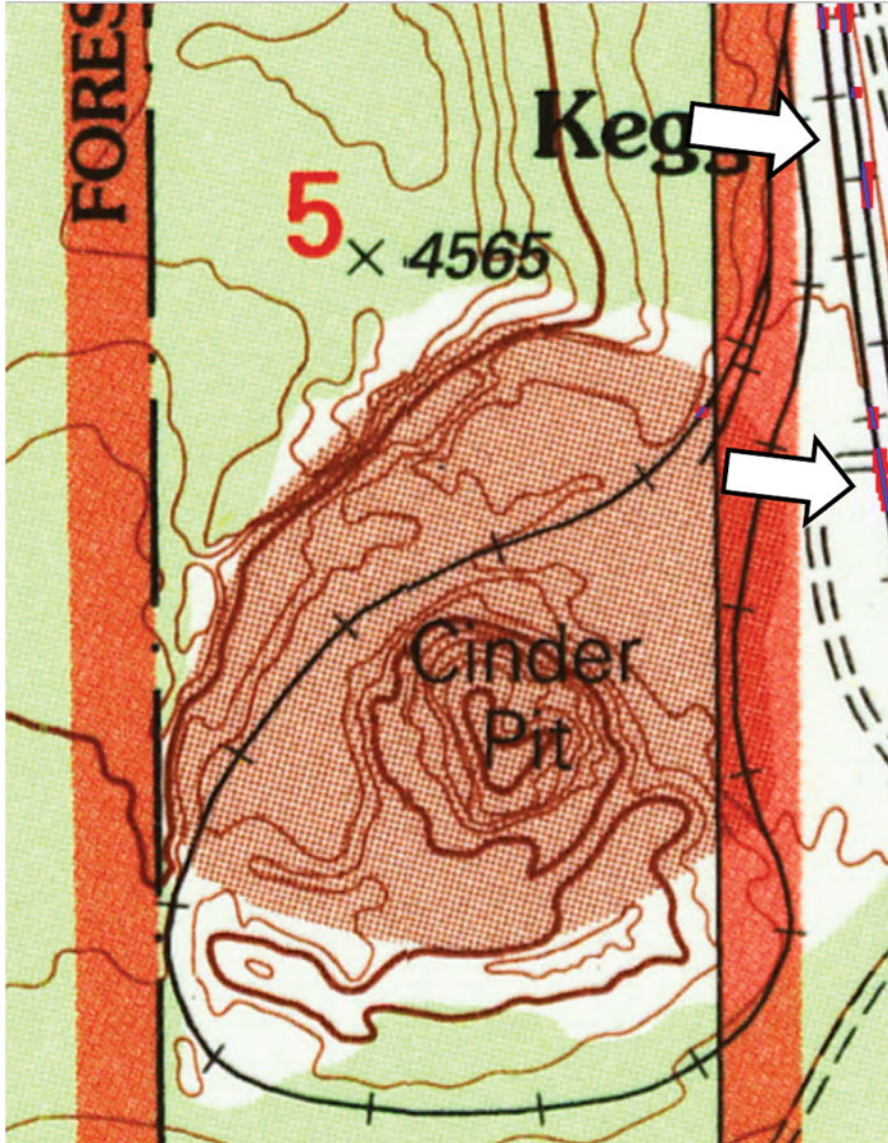


Fig. A.1 FCN-VGG16 semantic segmentation results overlaying with the test map (red: FP; blue: TP; black lines with crosses: railroads)

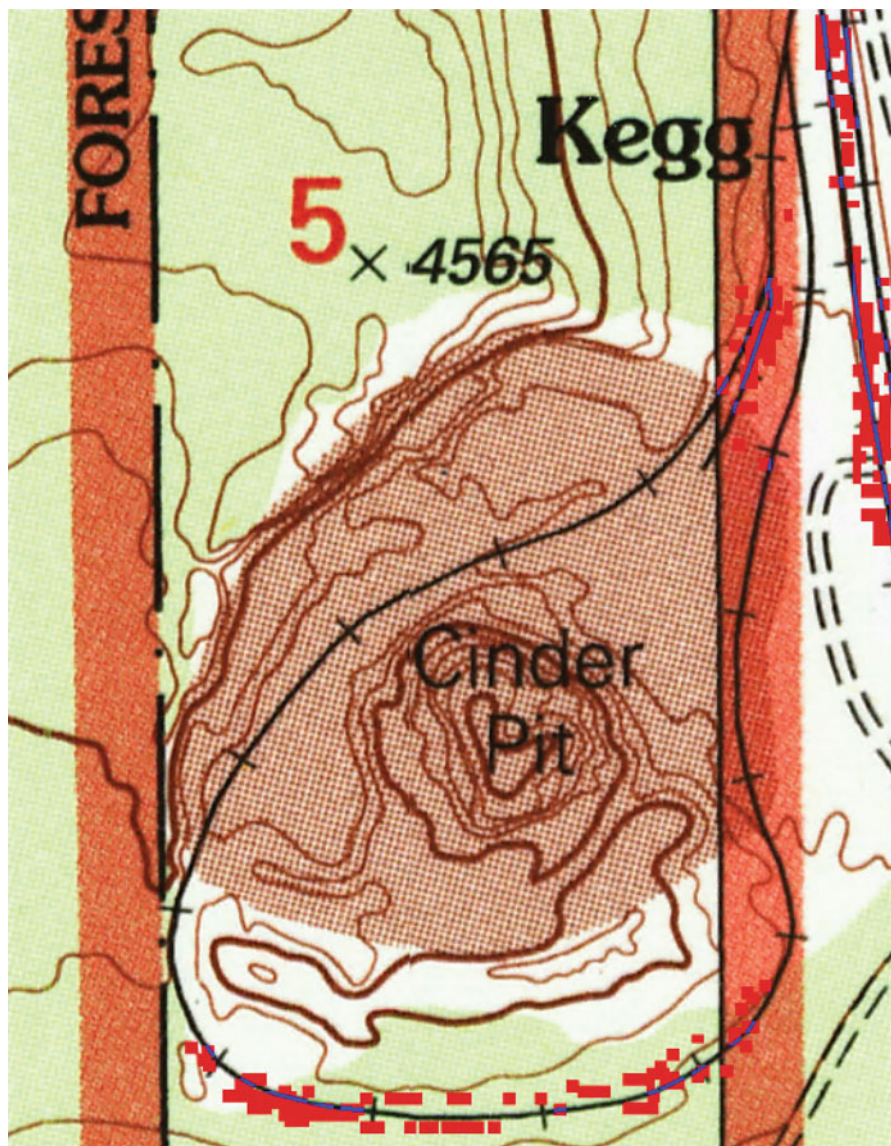


Fig. A.2 FCN-GoogLeNet semantic segmentation results overlaying with the test map (red: FP; blue: TP; black lines with crosses: railroads)

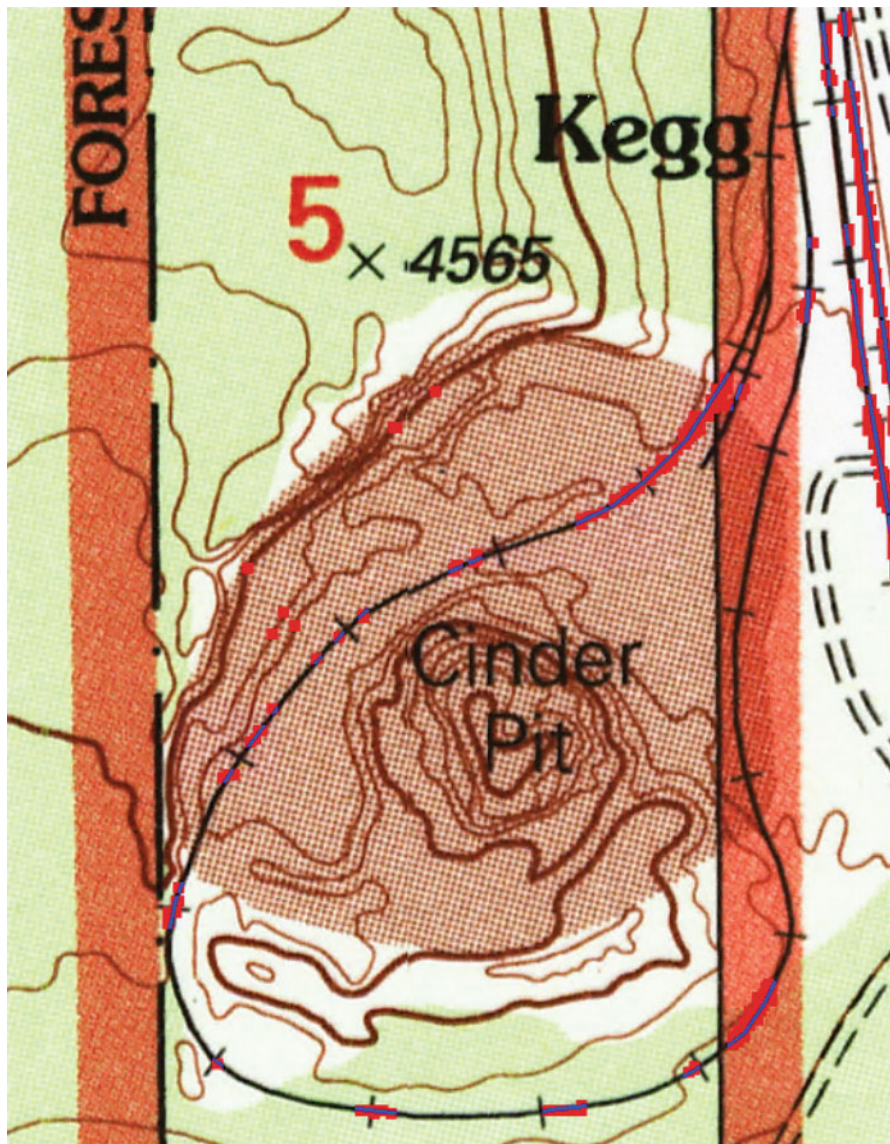


Fig. A.3 FCN-ResNet semantic segmentation results overlaying with the test map (red: FP; blue: TP; black lines with crosses: railroads)

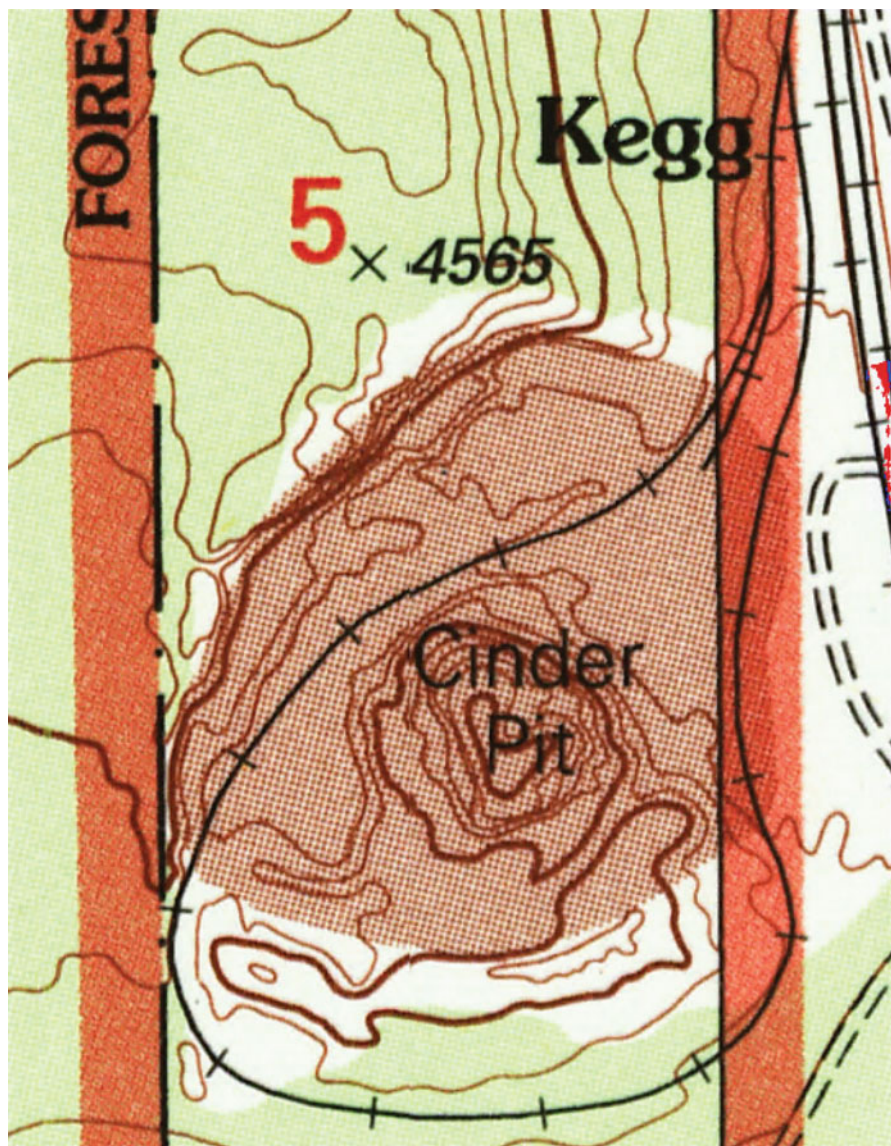


Fig. A.4 Semantic segmentation results (overlying with the test map) using pre-trained PSPNet with **small size training data** and **trained from the deep layer** (red: FP; blue: TP; black lines with crosses: railroads)

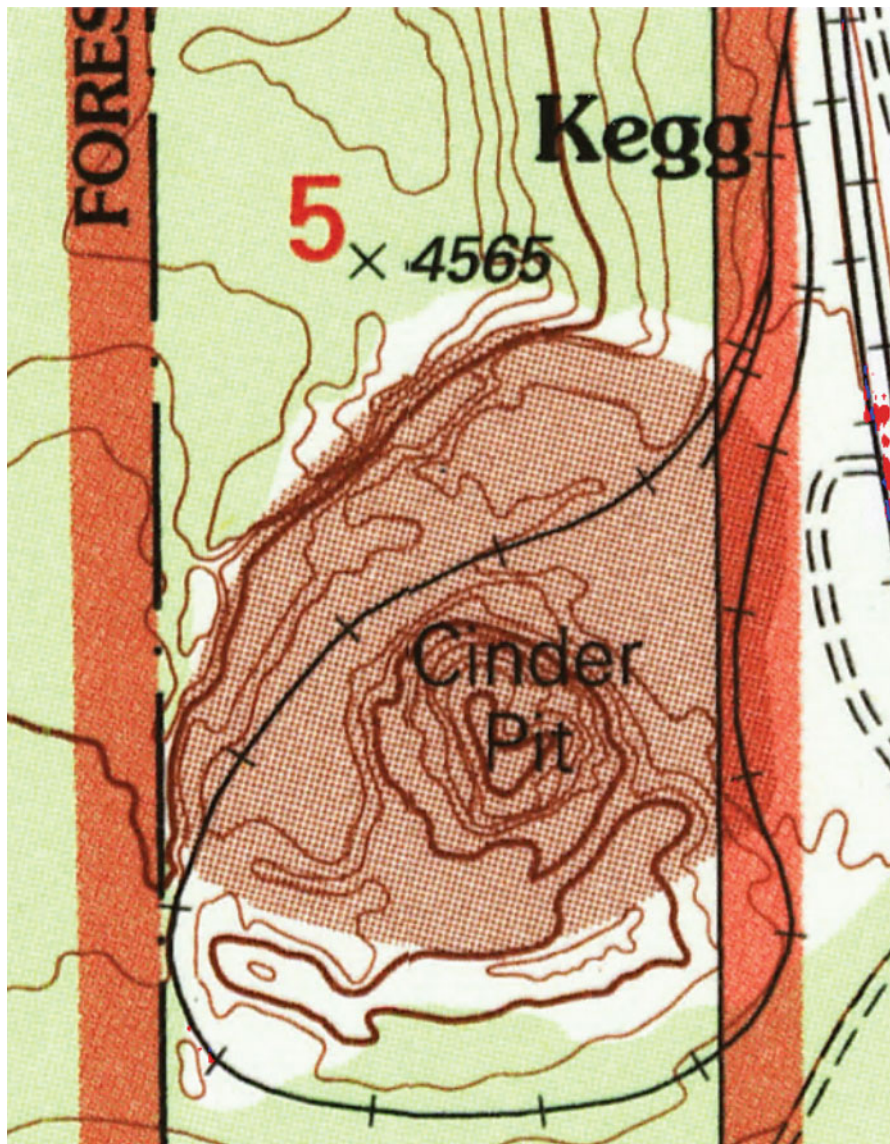


Fig. A.5 Semantic segmentation results (overlying with the test map) using pre-trained PSPNet with **small size training data** and **trained from the middle layers** (red: FP; blue: TP; black lines with crosses: railroads)

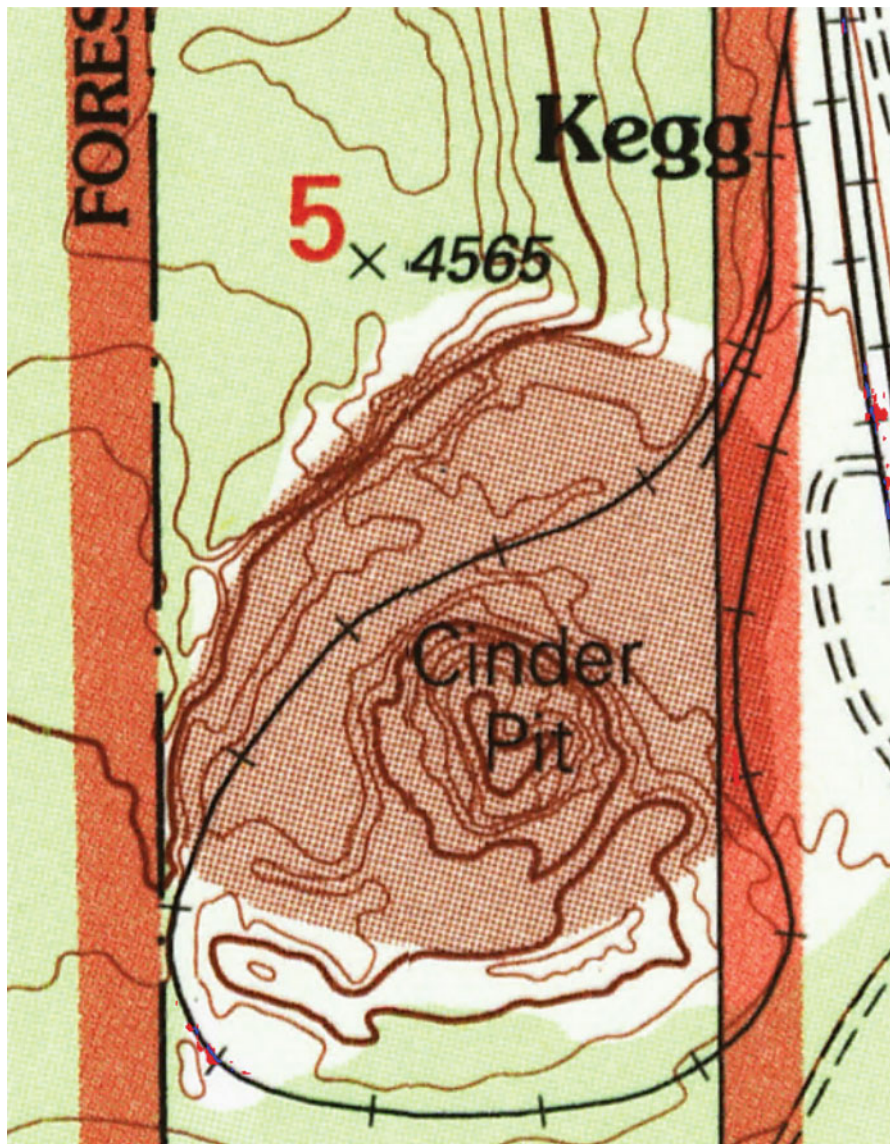


Fig. A.6 Semantic segmentation results (overlying with the test map) using pre-trained PSPNet with **small size training data** and **trained from the shallow layers** (red: FP; blue: TP; black lines with crosses: railroads)

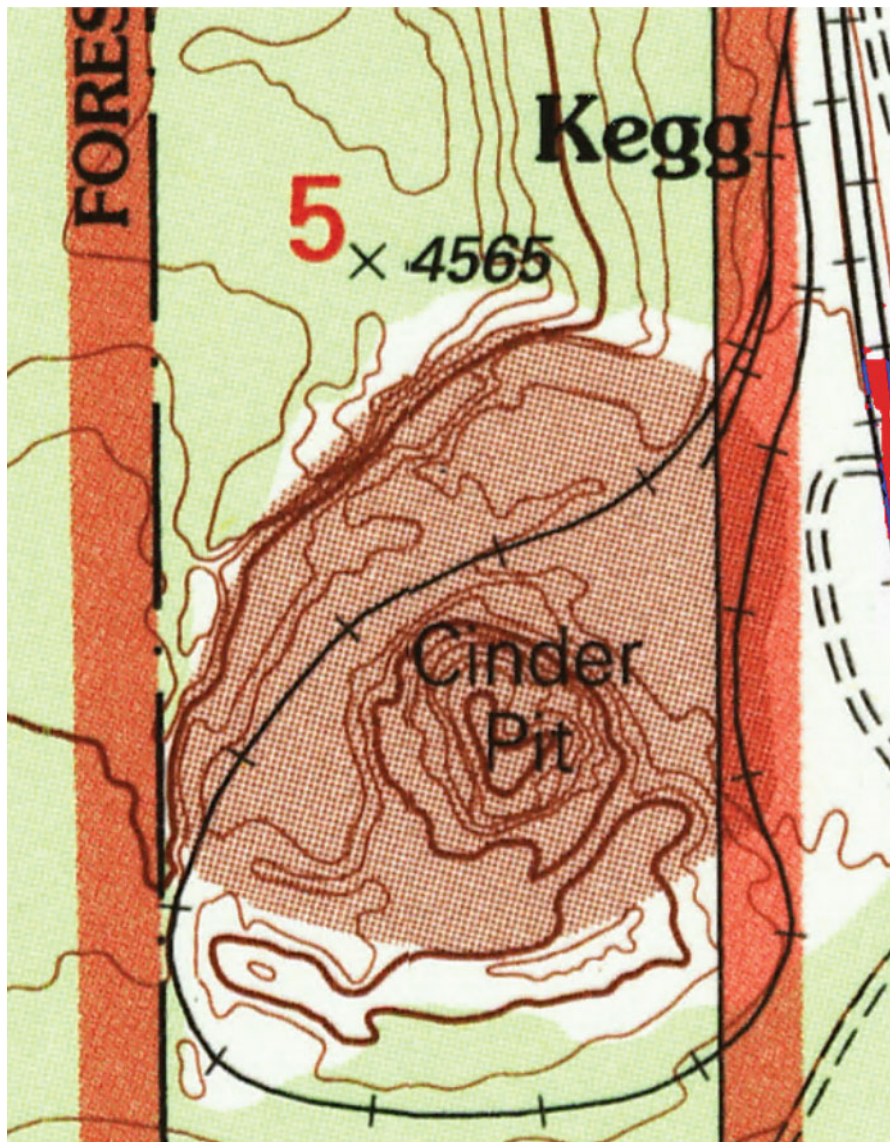


Fig. A.7 Semantic segmentation results (overlying with the test map) using pre-trained PSPNet with **medium size training data** and **trained from the deep layer** (red: FP; blue: TP; black lines with crosses: railroads)

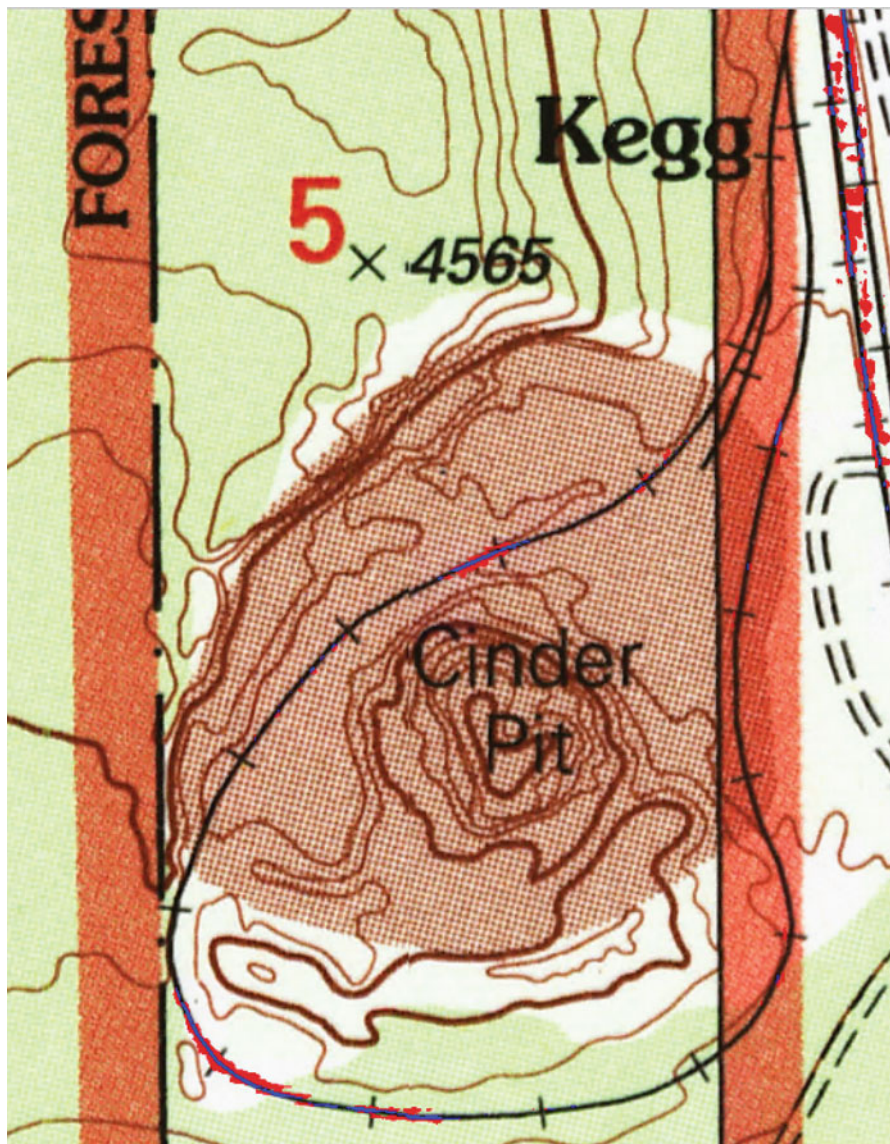


Fig. A.8 Semantic segmentation results (overlying with the test map) using pre-trained PSPNet with **medium size training data** and **trained from the middle layers** (red: FP; blue: TP; black lines with crosses: railroads)

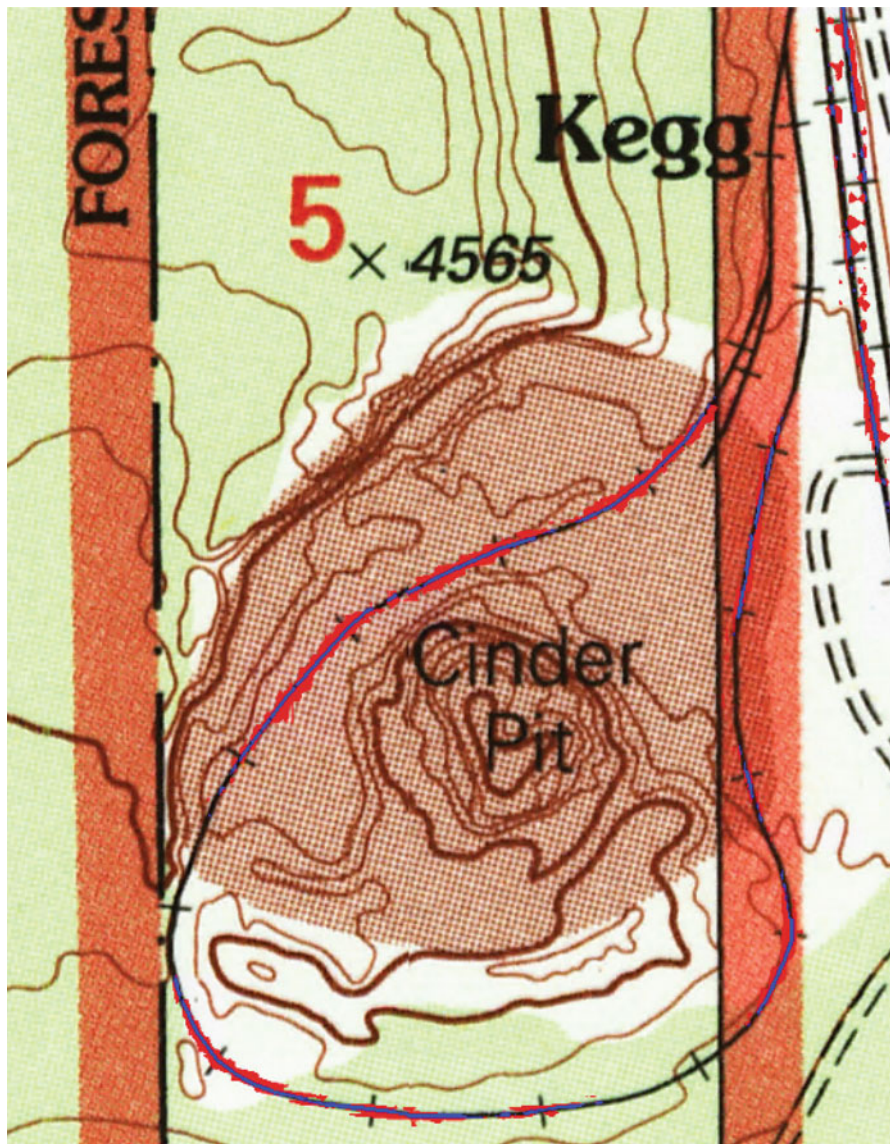


Fig. A.9 Semantic segmentation results (overlying with the test map) using pre-trained PSPNet with **medium size training data** and **trained from the shallow layers** (red: FP; blue: TP; black lines with crosses: railroads)

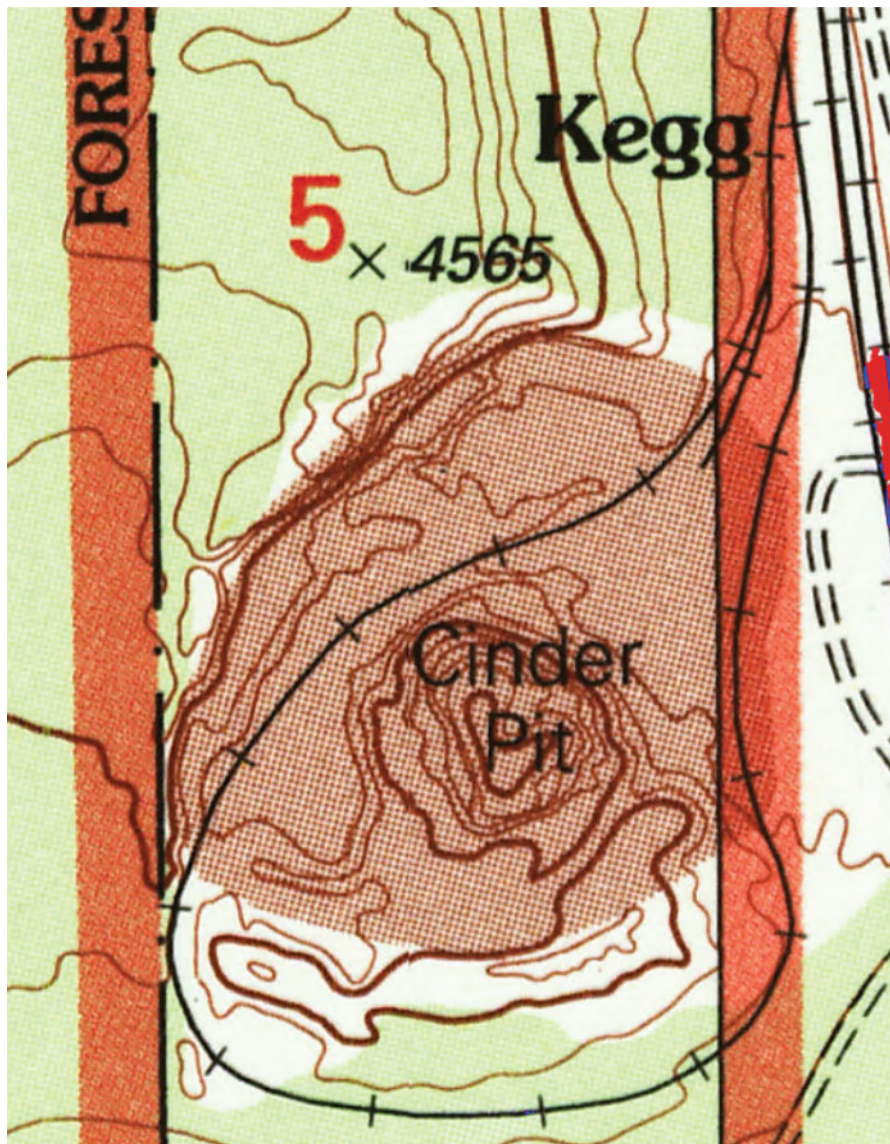


Fig. A.10 Semantic segmentation results (overlying with the test map) using pre-trained PSPNet with **large size training data** and **trained from the deep layer** (red: FP; blue: TP; black lines with crosses: railroads)

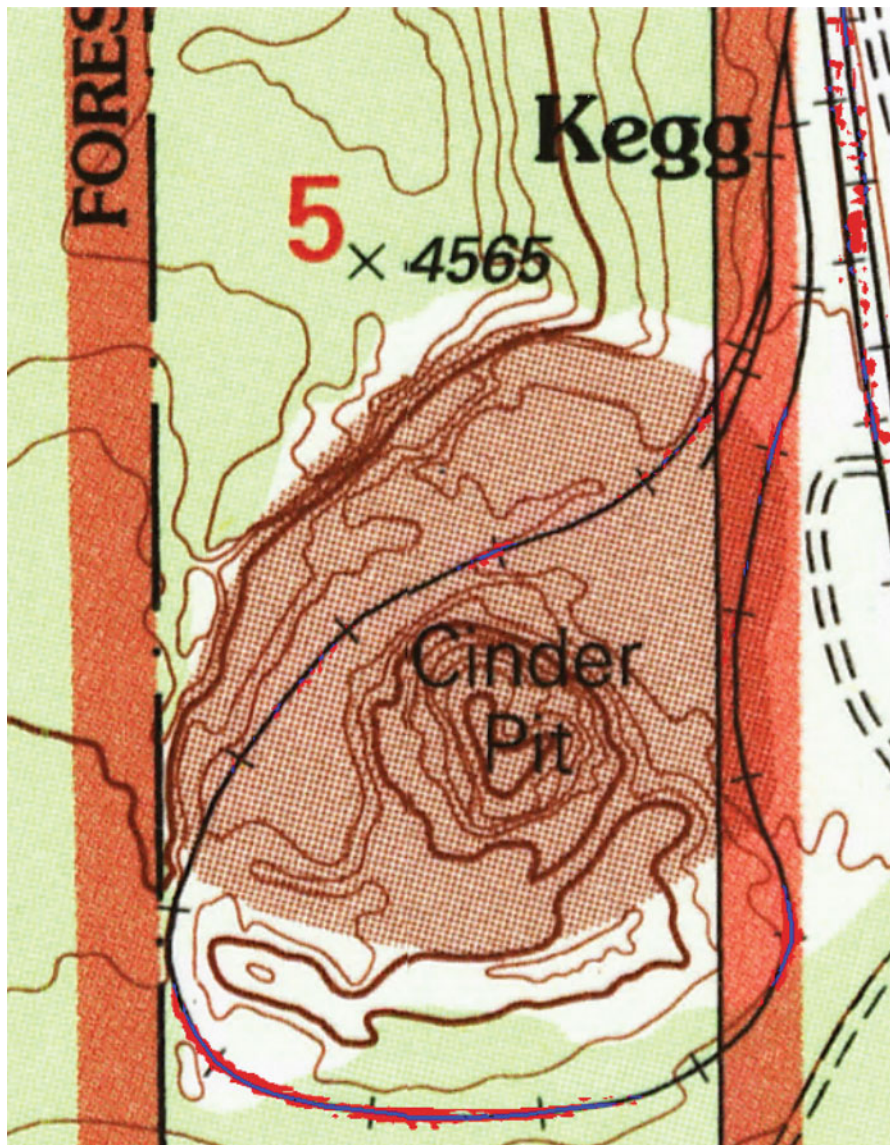


Fig. A.11 Semantic segmentation results (overlying with the test map) using pre-trained PSPNet with **large size training data** and **trained from the middle layers** (red: FP; blue: TP; black lines with crosses: railroads)

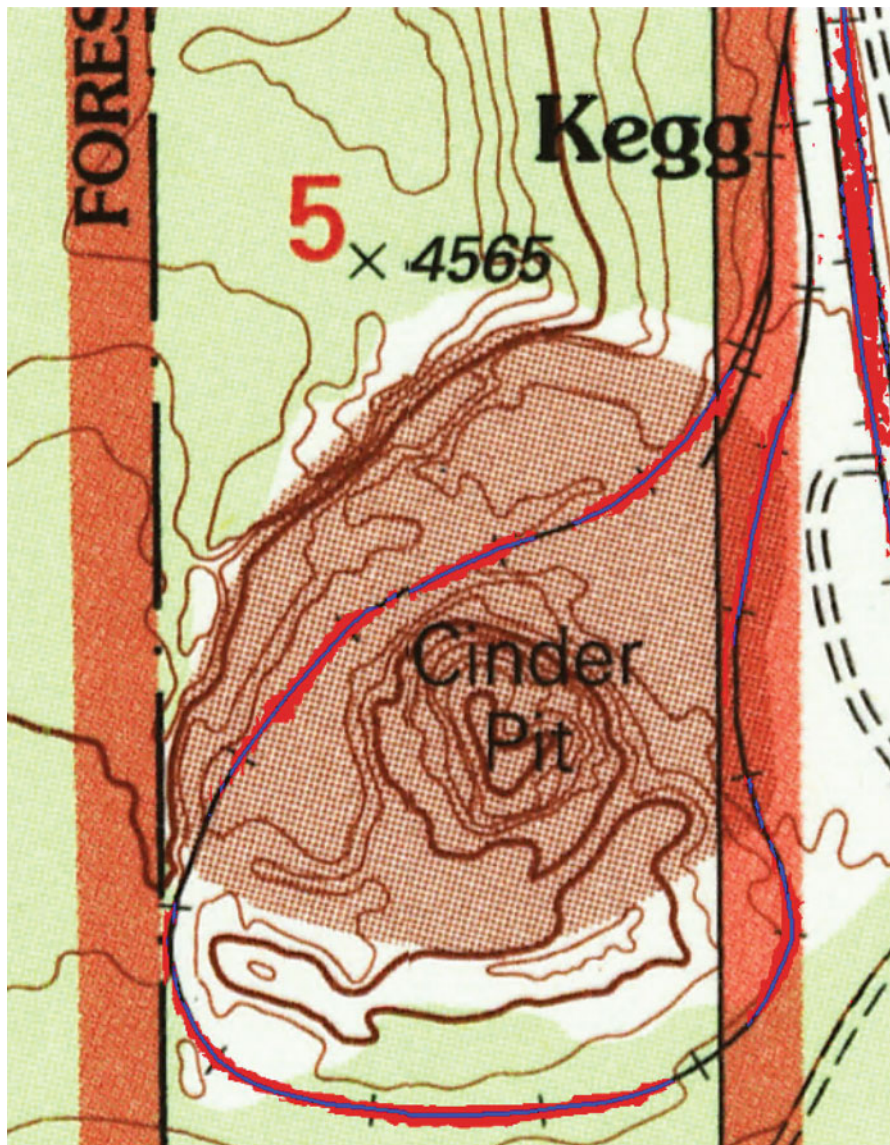


Fig. A.12 Semantic segmentation results (overlying with the test map) using pre-trained PSPNet with **large size training data** and **trained from the shallow layers** (red: FP; blue: TP; black lines with crosses: railroads)

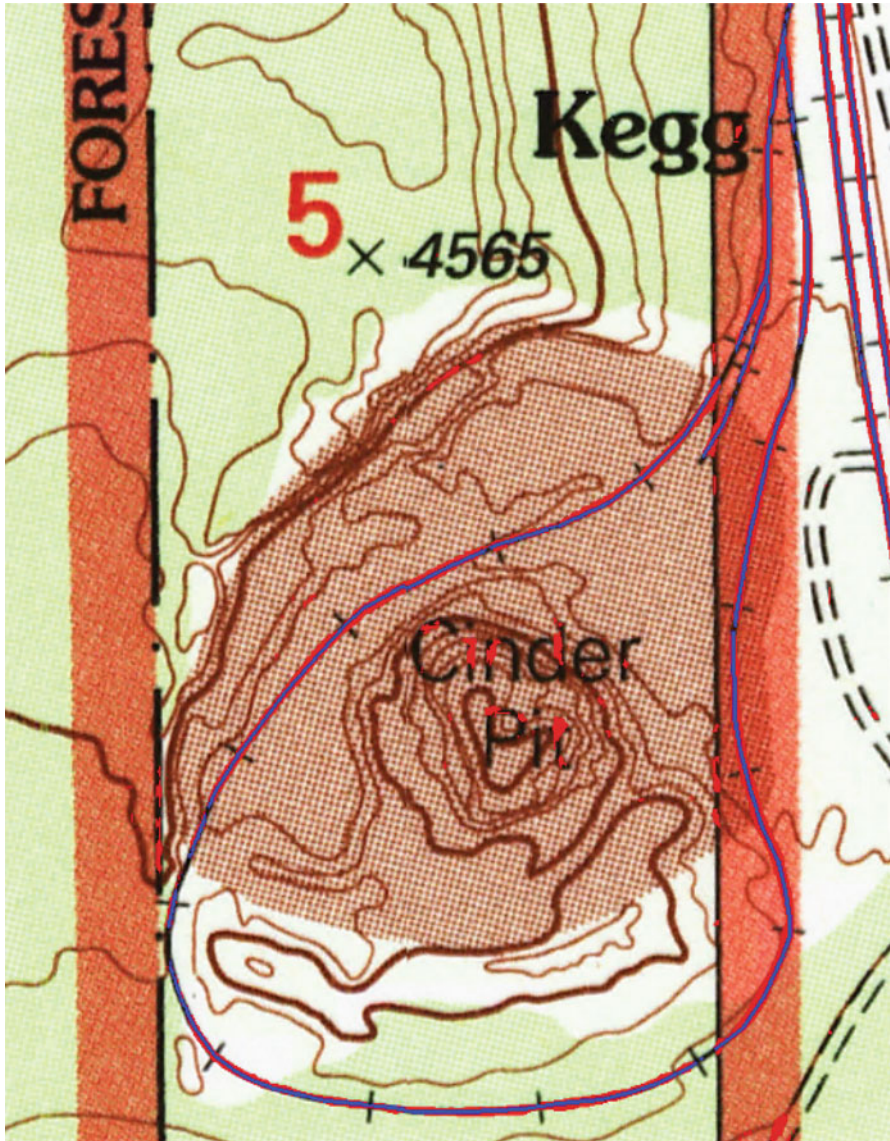


Fig. A.13 Semantic segmentation results (overlying with the test map) using the pre-trained, modified PSPNet **large size training data** and **trained from the shallow layers** (red: FP; blue: TP; black lines with crosses: railroads)