Automatically Annotating and Integrating Spatial Datasets

Ching-Chien Chen, Snehal Thakkar, Craig Knoblock, Cyrus Shahabi

Department of Computer Science & Information Sciences Institute,
University of Southern California,
Los Angeles CA 90089,
{chingchc, snehalth, knoblock, shahabi}@usc.edu

Abstract. Recent growth of the geo-spatial information on the web has made it possible to easily access a wide variety of spatial data. By integrating these spatial datasets, one can support a rich set of queries that could not have been answered given any of these sets in isolation. However, accurately integrating geo-spatial data from different data sources is a challenging task. This is because spatial data obtained from various data sources may have different projections, different accuracy levels and different formats (e.g. raster or vector format). In this paper, we describe an information integration approach, which utilizes various geo-spatial and textual data available on the Internet to automatically annotate and conflate satellite imagery with vector datasets. We describe two techniques to automatically generate control point pairs from the satellite imagery and vector data to perform the conflation. The first technique generates the control point pairs by integrating information from different online sources. The second technique exploits the information from the vector data to perform localized image-processing on the satellite imagery. Using these techniques, we can automatically integrate vector data with satellite imagery or align multiple satellite images of the same area. Our automatic conflation techniques can automatically identify the roads in satellite imagery with an average error of 8.61 meters compared to the original error of 26.19 meters for the city of El Segundo and 7.48 meters compared to 15.27 meters for the city of Adams Morgan in Washington, DC.

1. Introduction

Automatically and accurately aligning two spatial datasets is a challenging problem. Two spatial datasets obtained from different organizations can have different geographic projections and different type of inaccuracies. If the geographic projections of both datasets are known, then both datasets can be converted to the same geographic projections. However, the geographic projection for a wide variety of geospatial data available on the Internet is not known. Furthermore, converting datasets into the same projection does not address the issue of different inaccuracies between two spatial datasets. Despite the fact that GIS researchers have worked on this problem for a long time, the resulting conflation [22] algorithms still require the manual

identification of control points. Automatic conflation techniques are necessary to automatically integrate large spatial datasets. One application of automated conflation techniques is to accurately identify buildings in the satellite imagery. Computer vision researchers have been working on trying to identify features, such as roads, buildings, and other features in the satellite imagery [19]. While the computer vision research has produced algorithms to identify the features in the satellite imagery, the accuracy and run time of those algorithms are not suited for these applications.

We developed the Building Finder application, which integrates satellite imagery from Microsoft Terraservice with the street information from U.S. Census TIGER/Line files and building information from a white page web source to identify buildings in the satellite imagery. The Building Finder queries the streets from a database containing street network information. The result of the query is a set of tuples consisting of street name, city, state and zip code, which is used to query the Switchboard white pages agent to find the addresses related to those streets. The result of the Switchboard white pages website is then provided to the geocoder agent, which in turn provides the latitudes and longitudes for the addresses. The Building Finder also obtains a satellite image from Terraservice for the given area of interest. Finally, the latitude and longitude points representing different addresses and information representing different streets is superimposed on the satellite imagery.

A key research challenge in developing the Building Finder is to accurately integrate road network vector data with the satellite image. Different information sources utilize different projections for spatial information and there are various inconsistencies in the spatial information. For example, the spatial projection utilized for the satellite imagery is not the same as the spatial projection utilized for the TIGER/Line files, and due to local elevation changes some road locations in the TIGER/Line files are inaccurate. Due to these problems, finding accurate locations of the buildings in the satellite image is a very challenging problem. The Building Finder utilizes techniques described in this paper to find accurate locations of the buildings in the satellite image.

In this paper, our focus is on efficiently and completely automatically reducing spatial inconsistencies between two geo-spatial datasets originating from two different data sources. The spatial inconsistencies are due to the inaccuracy of different data sources as well as different projections used by different data source. Traditionally GIS systems have utilized a technique called conflation [22] to accurately align different geo-spatial datasets. The conflation process can be divided into the following subtasks: (1) find control point pairs in two datasets, (2) detect inaccurate control point pairs from the set of control point pairs for quality control, and (3) use the accurate control points to align the rest of the points and lines in both datasets using triangulation and rubber-sheeting techniques.

Applications, such as the Building Finder, cannot rely on a manual approach to perform conflation, as the area of interest for the Building Finder application may be anywhere in the continental United States. Manually finding and filtering control points for a large region, such as, the continental United States, is very time consuming and error-prone. Moreover, performing conflation offline on two datasets is also not a viable option as both datasets are obtained by querying different web sources at run-time. In fact, satellite imagery and vector data covering the whole world are

available from various sources. The vector data and the satellite imagery obtained from different sources do not always align with each other and manually finding control points for the entire world is a very daunting task. Therefore, an automatic approach to find accurate control point pairs in different geo-spatial datasets is required. Our experimental results show that using our algorithm, we can completely automatically align two geo-spatial datasets.

The remainder of this paper is organized as follows. Section 2 describes two different algorithms to automatically identify control point pairs in two geo-spatial datasets. Section 3 describes an algorithm to filter out inaccurate control point pairs from the automatically generated control point pairs. Section 4 describes a modified conflation process to align two geo-spatial data sets. Section 5 provides the results of utilizing our approach to identify road network in the satellite imagery. Section 6 discusses the related work. Section 7 concludes the paper by discussing our future plans.

2. Finding Control Points

A control point pair consists of a point in one dataset and a corresponding point in the other dataset. Finding accurate control point pairs is a very important step in the conflation process as all the other points in both datasets are aligned based on the control point pairs. Section 2.1 describes a technique to find control points by querying information from existing web services. Section 2.2 describes a technique to generate control points using localized image processing.

2.1 Using Online Data

The Internet has a wide variety of geo-spatial and textual datasets available on the web. Intuitively, the idea behind finding control points using the online data sources is to find some feature points on one of the datasets and utilize sources on the Internet to find the corresponding points on the second dataset. In case of the Building Finder, Microsoft Terraservice provides the satellite imagery dataset. Terraservice also provides different types of feature points, such as churches, buildings, schools, etc through the Terraserver Landmark Service. The points provided by Terraserver Landmark Service align perfectly with the satellite imagery, i.e. the points line up with corresponding features in the satellite imagery. Therefore, the feature points provided can be used as control points on the satellite imagery. The feature points extracted from the Terraserver Landmark Service provide name of the point, latitude and longitude for each point. One way to find corresponding point on the TIGER/Line files is to find the address of each feature point and geocode the addresses using the TIGER/Line files. However, the landmark feature points only provide name, type, and coordinates of the important points in various categories, such as churches, hospitals, etc. Table 1 shows some example landmark points queried from Microsoft TerraService.

As shown in Figure 1, the corresponding feature control points in the second dataset are identified by integrating information from several online sources. In case of

Table 1 TerraServer Landmark Feature Points

Feature Name	Type	Latitude	Longitude
Church of Christ	Church	33.91971	-118.4079
El Segundo Christian Church	Church	33.91811	-118.4179
El Segundo Public Library	Library	33.92391	-118.4169
El Segundo Foursquare Church	Church	33.92154	-118.4175
First Baptist Church	Church	33.92531	-118.4099

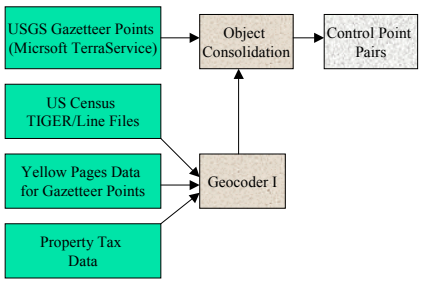

Figure 1 Finding Control Points Using Online Sources

the Building Finder, the second dataset is the TIGER/Lines vector data. The Building Finder queries the relevant yellow page web sources for the landmark points in various categories and finds a list of all points in the area for a category. We utilize machine learning techniques described in [17] to query web sources as if they are databases. Online yellow page sources are often incomplete or have some inaccuracies, so the Building Finder integrates information from the following yellow page web sources: (1) The Yahoo Yellow Pages, (2) The Verizon Superpages, and (3) The White pages. Next, we find the geographic coordinates (i.e., the latitude and longitude) of the addresses of the yellow page points using a geocoder that utilizes vector data from TIGER/Line files, i.e., the second dataset, to find geographic coordinates for the given addresses. This geocoded point provides the corresponding point on the TIGER/Line files. Some sample feature points identified by this method are shown in Table 2.

As shown in Table 1 and Table 2, the landmark names extracted from the yellow page sources do not exactly match with the landmark names from the Terraservice. Furthermore, different yellow page sources refer to different landmarks using different names. The Building Finder application utilizes the record linkage techniques [25] to identify matching point pairs from the landmark points obtained from Microsoft TerraService and the landmark features queried from different yellow page web sources. The record linkage techniques identify textual similarity between the records by utilizing different transformations, such as, acronym, substring, and stemming. The matching point pairs can be used as the control point pairs to conflate two data sources.

Figure 2 Resulting Control Points

The corresponding landmarks on both the imagery and vector data are good candidates for control point pairs. However, we must address the following challenges: First, the landmark points are not uniformly distributed on the imagery. Hence, there may not be enough landmark points in some areas to find sufficient control point pairs. Due to this problem, the available landmark points may not produce enough control point pairs to capture local transformations between the two geo-spatial datasets. We address this issue by utilizing a technique termed region growing, which is described in Section 4.3. Second, some landmarks are big entities that cover a large area. For example, a school may cover a rectangular area of 200 pixels width and 200 pixels height on a 1m/pixel resolution image, and the center of the school building is chosen as the representative for the landmark. The geocoder may geocode the point at the center of the rectangle, which would turn out to be different than the center of the building. We addressed this issue by utilizing small entities, like churches and police stations, as control points.

2.2 Analyzing Imagery Using the Vector Data

We also explored the use of image analysis to identify control point pairs. Various GIS researchers and computer vision researchers have shown that the intersection points on the road networks provide an accurate set of control point pairs [8, 10]. In fact, several image processing algorithms to detect roads in the satellite imagery have been utilized to identify intersection points in the satellite imagery. Unfortunately, automatically extracting road segments directly from the imagery is a difficult task due to the complexity that characterizes natural scenes [11]. Moreover, processing an image of a large area to extract roads requires a lot of time.

Integrating vector data into the road extraction procedures alleviates these problems. We developed a localized image processing technique that takes advantage of the vector data to accurately and efficiently find the intersection points of various roads on the satellite image. Conceptually, the spatial information on the vector data represents the existing knowledge about the approximate location of the roads and intersection points on the satellite imagery. We improve the accuracy and run time of the algorithms to detect intersection points in the satellite image by utilizing the knowledge from the vector data. First, our localized image processing technique finds all the intersection points on the vector data. For each intersection point on the vector data, the localized image processing technique determines the area in the satellite image where the corresponding intersection point should be located. Finally, the image processing techniques are applied to these small areas to identify the intersec-

Table 2 Extracted Feature Points from Online Sources

Feature Name	Address	Latitude	Longitude
Church of Christ El Segundo Hilltop Community	717 East Grand Ave	33.91961	-118.4079
El Segundo Christian Church	223 West Franklin Ave	33.91751	-118.4139
El Segundo Public Library	111 W Mariposa Ave	33.92331	-118.4159
Foursquare Church Of El Se-gundo	429 Richmond Street	33.92124	-118.4145
First Baptist Church of El Se-gundo	591 East Palm Avenue	33.92501	-118.4049

tion points on the satellite imagery. The area size of selected areas is much smaller than the entire image. The area is determined from the intersection points on the vector data and the directions of the road segments intersecting at these points.

The localized image processing technique may not be able to find all intersection points on the satellite image due to the existence of trees or other obstructions. However, the conflation process does not require a large number of control point pairs to perform accurate conflation. Therefore, for a particular intersection point on the vector data, if the corresponding image intersection point cannot be found within the certain area, it will not greatly affect the conflation process. We discuss the more detailed procedure in the following sub-sections.

2.2.1 Road Networks Intersection Detection

The process of finding the intersection points on the road network from the vector data is divided into two steps. First, all candidate points are obtained by examining all line segments in the vector data. In this step, the endpoints of each line segment in the vector data are labeled as the candidate points. Second, the connectivity of these candidate points is examined to determine if they are intersection points. In this step, each candidate point is examined to see if there are more than two line segments connected at this point. If so, this point is marked as an intersection point and the directions of the line segments that are connected at the intersection point are calculated.

2.2.2 Imagery Road Intersection Detection (Localized Image Processing)

The intersection points and the road directions from the vector data are utilized to identify the corresponding intersection points on the satellite image. The algorithm to identify intersection points in the imagery takes the following parameters: the satellite image, coordinates of the corner points of the satellite image, set of intersection points detected from the vector data, and the area size parameter. The area size determines the size of the rectangular area around the intersection point examined by our localized image processing algorithm. The area size parameter can be determined based on the accuracy of the two data sets. One option is to utilize the maximum error or offset between two datasets. We utilized the information from the US Census Bureau survey [16] to determine the area size parameter. The area size parameter can also be estimated by the following incremental procedure: First, randomly pick an intersection point on the vector data. Next, mark the location in the image at the same coordinates as the intersection point from the vector data. Start with a very small area size and gradually increase the area size until some clear linear features within the area are recognized. Note the value of the area size parameter. Repeat this procedure for a few intersection points and pick the maximum area size.

For each intersection point detected from the vector data, the localized image processing technique picks a rectangular area in the satellite image centered at the location of the intersection point from the vector data. The existing edge detection techniques from [18] are used to identify linear features in the area. An accumulation array [21] technique is utilized to detect line segments from the linear features. The detected linear features and directions of the lines from the vector data are the key variables used to determine the score for each linear feature (on the imagery) in the accumula-

tion array. The line segment formed by the images' linear features with the highest score in the accumulation array pinpoints the location of the edges of the roads. The intersection point of the detected lines is most likely the corresponding intersection point on the satellite imagery.

The localized image processing avoids exhaustive search of all intersection points on the entire satellite image and often locates the intersection point on the satellite image that is the closest intersection point to the intersection point detected from the vector data. Moreover, this technique does not need to extract road segments for the entire region. Only partial road segments near the intersections on the satellite image need to be extracted. Extracting road segments near the intersection point is easier than extracting all road segments, as the road sides closest to the intersections are often two parallel strong linear features, which are easier to identify. Figure 3 depicts the intersection points on vector data and the corresponding intersection points on imagery. The rectangular points are the intersection points on the vector data, and the circular points are the intersection points on the images.

3. Filtering Control Points

Both techniques discussed in Section 2 may generate some inaccurate control point pairs. As discussed in Section 2.1, the approach to identify control point pairs using online data sources may produce inaccurate control point pairs due to temporal inconsistencies between data sources and the size of the various features. Meanwhile the localized image processing may identify linear features, like tree clusters, building shadings, building edges and some other image noise, as road segments, thus detecting some inaccurate control point pairs. For example in Figure 3, the control point pairs 1, 2 and 3 are inaccurate control point pairs.

The conflation algorithm utilizes the control point pairs to align the vector data with the satellite image. The inaccurate control point pairs reduce the accuracy of the alignment between two datasets. Therefore, it is very important to filter out inaccurate control point pairs. While there is no global transformation to align imagery and vector data, in small areas the relationship between the points on the imagery and the points on the vector data can be described by a transformation and a small uncertainty measure. The transformation segment can be attributed to different projections used to obtain the imagery data and the vector data, while the small uncertainty measure is due to the elevation changes in the

Figure 3. The intersection points (rectangles) on vector data and the corresponding intersection points (circles) on imagery

area or due to the inconsistencies between the datasets. Due to the above-mentioned nature of the datasets, in a small region the control points on the imagery and the counterparts on vector data should be related by similar transformations. Therefore, the inaccurate control point pairs can be detected by identifying those pairs with significantly different relationship as compared to the other nearby control point pairs. We used the vector median filter (VMF) [1] to filter out inaccurate control points.

3.1 Vector Median Filter (VMF)

Vector Median Filter (VMF) [1] is a mathematical tool for signal processing to attenuate noise, and it is a popular filter to do noise removal in image processing. The VMF accepts the data points as vectors, e.g., in our case a 2D vector with latitude and longitude differences between the points in the control point pair, and filters out the data points with the vectors significantly different from the median vector.

The geographic coordinate displacement between the points of each control point pair in a small area can be viewed as a 2D vector, termed control-point vector. The starting point of the vector is the control point on the vector data and the end point is the control point on the image. Because the spatial inconsistencies between the imagery and vector data in a local area are similar, the control-point vectors whose direction and length are significantly different from the others are characterized as an inaccurate control-point vector. Due to the similarities of these control-point vectors, the directions and lengths of them can be represented by the vector median. We modified the vector median filter to assist us in identifying the control-point vectors that are significantly different. This helped to obtain the best matching set of control points.

Vector median has similar properties as the median operation. Intuitively, the median vector is the vector that has the shortest summed distance (Euclidean distance) to all other vectors.

The inputs for a vector median filter are N vectors $\vec{x}_i$ (i= 1, 2, 3, … N) and the output of the filter is the vector median $\vec{x}_{vm}$. We revised the output of vector median filter to accommodate not only $\vec{x}_{vm}$, but also k closest vectors to the vector median. We defined the distance D:

$$D = \| \vec{x}_k - \vec{x}_{vm} \|_2$$

where $\vec{x}_k$ is the k-th closest vector to $\vec{x}_{vm}$. Then, the output of our vector median filter is

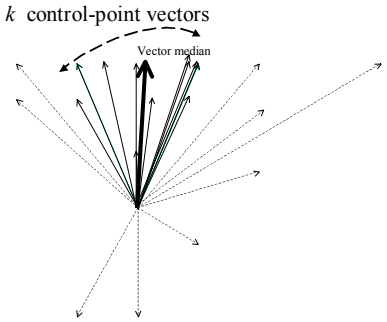

Figure 4. The distributions of twenty-one control-point vectors in Figure 3(k =11).

$$\{ \vec{x}_i \mid \text{where } \| \vec{x}_i - \vec{x}_{vm} \| \le D \}$$

As shown in Figure 4, the modified Vector Median Filter selects the k closest vectors to the vector median as the accurate control point pairs. The possible value of k is an integer between 1 and N. Large value of k provides more control-point vectors, but may not filter out all inaccurate control point pairs. If the number of inaccurate control point pairs exceeds the half of the size of control-point pairs, then the vector median would be one of the inaccurate vectors. The Vector Median Filter can only work when the median vector is not inaccurate. Thus, the number of inaccurate control-point vectors should not exceed half the control-point vectors. Therefore, control point pairs with the $\left\lceil \frac{N}{2} \right\rceil$ closest vectors to the vector median should be the most accurate control point pairs. Towards this end, we kept the $k=\left\lceil \frac{N}{2} \right\rceil$ closest vectors to the vector median and filtered out the rest of the control point pairs. As a result, some accurate control-point vectors may be lost. However, the missing control point pairs would not greatly affect the conflation results, as some of the selected control point pairs close to the lost accurate control point pairs have similar directions and displacements.

4. Conflating Imagery And Vector Data

After filtering the control point pairs, we obtain accurate control point pairs on imagery and vector data. Each pair of corresponding control points from the two datasets indicates identical positions on each datasets. Transformations are calculated from the control point pairs. Other points in both datasets are aligned based on these transformations. The Delaunay Triangulation [5] and piecewise linear rubber sheeting [28] are utilized to find the appropriate transformations. The Delaunay Triangulation is discussed in Section 4.1, and rubber-sheeting is explained in Section 4.2. Moreover, a novel technique to alleviate the spatial inconsistencies for those areas where we cannot exploit any control point pairs from either techniques, is discussed in Section 4.3.

4.1 Triangulation

To achieve overall alignment of imagery and vector data, vector data must be adjusted locally to conform to the imagery. It is reasonable to align the two datasets based on local adjustments, because small changes in one area should not affect geometry at longer distances. To accomplish local adjustments, the domain space is partitioned into small pieces. Then, local adjustments are applied on each single piece. Triangulation is an effective strategy to partition domain space to define local adjustments.

There are different triangulations for the control points. One particular triangulation, the Delaunay triangulation, is especially suited for conflation systems [22]. A Delaunay triangulation is a triangulation of the point set with the property that no point falls in the interior of the circumcircle of any triangle (the circle passing

through the three triangle vertices). The Delaunay triangulation maximizes the minimum angle of all the angles in the triangulation, thus avoiding triangles with extremely small angles. We perform the Delaunay triangulation with the set of control points on the vector data, and make a set of equivalent triangles with corresponding control points on the imagery. The Delanuay triangulation can be built in O(n*log n) time in worst case, where n is the number of control points.

4.2 Piecewise Linear Rubber-sheeting

Imagine stretching a vector map as if it was made of rubber. We deform the vector data algorithmically, forcing registration of control points over the vector data with their corresponding points on the imagery. This technique is called "Piecewise linear rubber sheeting" [28]. There are two steps to rubber sheeting. First, the transformation coefficients to map each Delanuay triangular on vector data onto its corresponding triangular on the imagery are calculated. Second, the same transformation coefficients are applied to the road endpoints inside each triangle to transform the road endpoints (on the vector data) within the triangle. The conflated road network is constructed from these transformed endpoints.

Piecewise linear rubber sheeting based on triangles with extremely small angles (i.e., long and thin triangles) results in distorted conflation lines. Since the Delanuay triangulation avoids triangles with extremely small angles, it alleviates the problem. The details of the triangulation techniques and the piecewise linear rubber-sheeting algorithms are described in [15, 22, 28].

4.3 Region Growing

We propose a technique named "region-growing" to alleviate the spatial inconsistencies for those areas where there are no feature points to perform conflation (such as landmarks or intersection points) on the vector data and imagery. New control points are obtained by extrapolating existing control points. Using these new control points, the region with the control points can be expanded. This can also save time by reducing the need to detect intersection points or landmarks. However, if the existing control points are not accurate, the new control points will not be accurate either. In practice, "region-growing", "control points from online data sources" and "control points from intersection detections" could be combined to generate new control points for conflation.

Figure 5 illustrates the vector data for some streets in the city of El Segundo before conflation. Figure 6 shows the road network after applying our conflation technique, using VMF-filtered online data sources as control point pairs. Figure 7 shows the road network after applying conflation technique, using VMF-filtered intersection points as control point pairs.

Figure 5. The road network before conflation.

Figure 6. After applying conflation, utilizing VMF-filtered online data sources

Figure 7. After applying conflation, utilizing VMF-filtered intersection points

5. Performance Evaluation

We evaluated our approaches to accurately integrate different geo-spatial datasets by integrating data from two different datasets. The first dataset was the vector data (road networks), and the second dataset was satellite imagery. These datasets are described in detail in section 5.1. The purpose of the integration experiment was to evaluate the utility of these algorithms in integrating real world data. We are interested in evaluating the two approaches to generate the control point pairs and the effect of the filtering techniques. Moreover, we were interested in measuring the improvement in the accuracy of the integration of two datasets using our techniques. To that end, we performed experiments to validate the following:

Hypothesis 1: Performing automated conflation using the automated control point identification techniques described earlier (with no filters) improves the accuracy of the road identifications.

Hypothesis 2: The automated filtering techniques improve the accuracy of the road identifications for both automated control point identification techniques.

Hypothesis 3: The combination of the localized image processing using intersection points and the modified Vector Median Filter provides the best results.

Section 5.1 describes the experimental setup and the datasets used to evaluate our methods. Section 5.2 discusses performance

of the two automatic control point identification algorithms without any filters. Section 5.3 describes the improvement due to the Vector Median Filter.

5.1 Experimental Setup

The following are two different datasets used for our experiments: (1) Satellite imagery: The satellite imagery used in the experiments is the geo-referenced USGS DOQ images with 1-meter per pixel resolution. Microsoft TerraService web service [2, 3] was utilized to query the satellite imagery for different areas and (2) Vector data (road networks): The road network from the TIGER/Line files [26] was used as the vector data. The TIGER/Line files dataset was developed by the U.S. Bureau of the Census. In general, the TIGER/Lin files dataset has richer attribution but poor positional accuracy. As shown in Figure 5, the road network is TIGER/Line files and there are certain spatial inconsistencies between the satellite imagery and TIGER/Line files.

The automatic conflation system was developed in C#. The output of our conflation algorithm was a set of conflated roads for the TIGER/Line files. The experiment platform is a Pentium III 700MHz processor with 256MB memory on Windows 2000 Professional (with .NET framework installed). Our experiments were done on the City of El Segundo, California, and a region of the city of Adams Morgan, District of Columbia. We obtained similar conflation performance for these two cities. Therefore, in the following sub-sections, we will take the city of El Segundo as an example to explain our conflation results, and list the conflation result of city of Adams Morgan for reference. The experiments on the city of El Segundo covered the area with latitude from 33.916397 to 33.93095, and longitude from -118.425117 to -118.370173. It is a region of 5.2Km by 1.6Km (a 5200x1600 image with 1m/pixel ground resolution). There are approximately 500 TIGER/Line segments (i.e. about 500 endpoints) on this region. The Adams Morgan data covers a 2.8Km by 2.4Km rectangular area with corner points latitude and longitude (-77.006, 38.899) and (-76.974, 38.879) and contains 300 road segments. Most roads in both the cities are 15 to 30 meters in width. Both automated conflation techniques are order of magnitude faster compared to the other computer vision algorithms to detect features in the satellite imagery.

In order to evaluate our approaches, we compared the conflated roads with the accurate roads. The accurate roads were generated by conflating the TIGER/Line data using the control point pairs provided manually. The road endpoints on the imagery were represented by the endpoints of the high accuracy road networks. The experiments used all the road endpoints in the conflated data and measured the displacement of the road endpoints compared to the corresponding road endpoints in the accurate road network. The mean and the standard deviation of the point displacements are used to evaluate the accuracy of the algorithms.

5.2 First Set of Experiments: Online Data Vs. Intersection Points

In the first set of experiments, we compared the mean standard deviation, and displacement range of the manually conflated road network with the road network generated using online data and intersection points as control point pairs respectively.

The experimental results are listed in Table 3, and the displacement distributions of the conflated roads' endpoints are shown in Figure 8. The X-axis of this Figure depicts the displacement between endpoint on the conflated roads and the equivalent endpoint on satellite image. The displacement values are grouped every 5 meters. The Y-axis shows the percentage of conflated points that are within the displacement range represented by the X-axis. For example, as shown in Figure 8, when utilizing unfiltered intersection points to generate conflated roads, 48% of the conflated roads' endpoints have less than 10 meters displacement from the corresponding satellite imagery points. While utilizing unfiltered online data, we obtained 13% of the points within 10 meters displacement. Considering the original TIGER/Lines, there are no points within 10 meters displacement from the imagery.

As shown in Table 3, the method utilizing unfiltered intersection points resulted in a smaller mean displacement than the TIGER/Lines and the conflated roads generated by the unfiltered online data. Therefore, the automated conflation approach using the control pairs obtained by using the localized image processing technique with no filter improves the accuracy of the integration process. However, the automated conflation using the unfiltered online control point pairs, resulted in lower accuracy as compared to

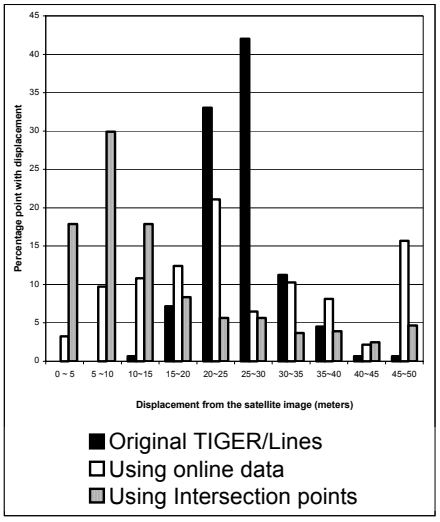

Figure 8. The displacement distributions of road endpoints (online data vs. intersection points) for city of El Segundo

Table 3 Comparison of original road network with conflated roads

Dataset	Mean point displacement	Standard Deviation	Mean +- std. deviation
Original TIGER/Lines El Segundo	26.19	5	(21.19, 31.19)
Online control points El Segundo	27.41	16.25	(11.16,43.66)
Intersection control points El Segundo	15.48	13.41	(2.07, 28.89)
Original TIGER/Lines Adams Morgan	15.27	3.31	(11.96, 18.58)
Intersection control points Ad-ams Morgan	11.74	9.71	(2.03, 21.45)

the original TIGER/Lines. The key reasons for the inaccurate results are that the landmark points obtained from the online sources are not uniformly distributed and the control point pairs are often inaccurate (because of the spatial inconsistencies between the online data sources). The inaccuracy of the control point pairs is accumulated when applying region growing to generate new conflated roads, resulting in almost half (44%) of the conflated points having greater than 25m displacement as shown in Figure 8. However, the performance of both approaches is significantly improved by filtering out the inaccurate control point pairs.

5.3 Second Set of Experiments: Filtered Control Points Vs. Unfiltered Control Points

In the second set of experiments, we utilized the Vector Median Filter to filter out inaccurate control point pairs from the control point pairs generated using online data sources or localized image processing. We identified the road network in the satellite imagery using the filtered control point pairs. Finally, the conflated roads were compared with the manually conflated road network to evaluate their performance.

The experimental results are listed in Table 4, and the displacement distributions of the conflated road endpoints are shown in Figure 9. The meanings of X-axis and Y-axis of Figure 9 are the same as Figure 8. As shown in Table 4, conflated lines using VMF-filtered online control point pairs increase the accuracy of the original data by about 40%. Moreover, the intersection control points with the Vector Median Filter leads to a displacement error that is less than 50% of the displacement error for the original vector data.

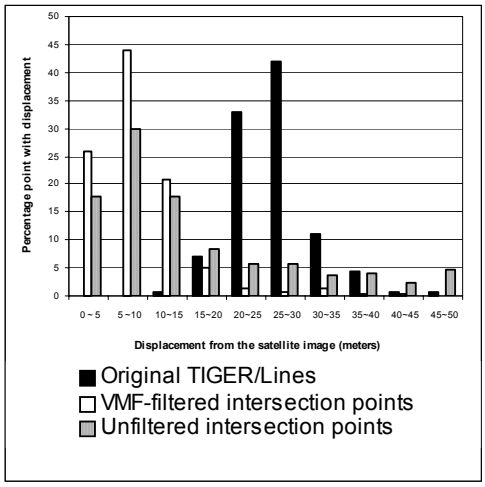

Figure 9. The displacement distributions of road endpoints (VMF-filtered vs. unfiltered intersection points) for city of El Segundo

From Figure 9, we can see

Table 4 Results after filtering

Dataset	Mean displacement	Standard deviation	Mean +- std. deviation (meters)
Original TIGER/Lines for El Segundo	26.19	5	(21.19, 31.19)
Intersection cps + VMF-filter for El Segundo	8.61	6	(2.61,14.61)
Online cps + VMF-filter for El Segundo	15.92	8.38	(7.54, 24.3)
Original TIGER/Lines for Adams Morgan	15.27	3.31	(11.96, 18.58)
Intersection cps + VMF-filter for Adams Morgan	7.48	4.81	(2.67, 12.29)

that when using VMF-filtered intersection points as control points, more than 60% of the conflated road endpoints are within 10 meters displacements from the image. Only 2.8% of the endpoints have displacements greater than 25 meters. After visual checking, we found most of these points are close to the margins of our experiment region. It is reasonable to have low accuracy points around the margins, since long and thin Delanuay triangles were constructed around the margins. The small value of standard deviation (6 meters) of conflated roads generated using VMF-filtered intersection points indicates that most points' displacements are close to the mean displacement of 8.6m. Although the standard deviation is one meter greater than the standard deviation of the TIGER/Lines (5 meters) data, the range of displacement from the image is much smaller than the TIGER/Lines' range of displacement. This means that majority of endpoints of conflated roads using the VMF-filtered intersection control points are more accurate than TIGER/Lines' endpoints.

From Table 3 and 4, we conclude that all methods to perform automated conflation, except the method utilizing unfiltered online data, result in more accurate alignment of the vector data with the satellite imagery and more accurate road identifications compared to the original road network. Therefore, using any combination of the automatic control point identification techniques and the automatic filters results in better alignment. This validates hypothesis 2.

From Table 4, we also see that the mean displacement of conflated roads utilizing VMF-filtered intersection points is three times better than the original TIGER/Lines and almost two times better than the result without using the filter. Finally, conflation using intersection control point pairs and the VMF filter provides the most accurate result. This validates hypothesis 3.

6. Related Work

Currently, there are commercial products that utilize conflation techniques to provide integrated geospatial data. For example, NEXUS [20] was proposed by Nicklas to serve as an open platform for spatially aware applications. Since all kinds of spatial data can be integrated into the NEXUS system, it is a vital prerequisite that identical spatial objects from different datasets be matched in advance. Toward this end, the conflation technique discussed in [27] was applied to accomplish vector to vector dataset integration in the NEXUS system. Yuan and Tao proposed a componentware technology to develop conflation components and they demonstrated their approach for vector-to-vector conflation [29]. A commercial conflation product, MapMerger [6], also performs vector-to-vector conflation with limited human intervention to consolidate multiple vector datasets.

Advances in satellite imaging technology are making it possible to capture geospatial imagery with ever increasing precision. Remotely sensed images from space can offer a resolution of one meter or better. Utilizing imagery to vector conflation, this accurate imagery can assist in updating the relatively poor positional accuracy but rich attribution vector datasets, such as TIGER/Lines. To perform imagery to vector conflation, some spatial objects must be extracted from imagery to serve as control points. However, autonomous extraction of spatial objects from satellite imagery is a

difficult task due to the complexity that characterizes natural scenes. Various approaches were developed over the past few years to automatically or semi-automatically conflate imagery and vector data covering the overlapping regions. Most of these approaches detect the counterpart elements on the datasets, then apply traditional conflation algorithm (i.e. establishing the correspondence between the matched entities and transforming other objects accordingly) [9, 21, 23, 24]. These approaches are different, because of the different methods utilized for locating the counterpart elements.

Some approaches directly extract the features from imagery and convert them to vector format, then apply the typical map-to-map [12, 22] or linear conflation algorithm [7]. Extracting features directly from imagery and converting to vector format is a tough task. Taking the road extraction as an example, there exist many algorithms for extracting roads utilizing the characteristics of roads as prior knowledge [9, 21, 23, 24], while none of them give good results in all circumstances [11, 13] and most of them are time-intensive.

Other alternative approaches utilize existing vector databases as part of the prior knowledge. Integrating existing vector data as part of the spatial object recognition scheme is an effective approach. Vector data represents the existing prior knowledge about the data, thus reducing the uncertainty in identifying the spatial objects in imagery. Hild and Fritsch [14] processed vector data to extract vector polygons and performed image segmentation on imagery to find image polygons. Then, a polygon matching (or shape matching) algorithm is applied on both images and vector to find a set of 2D conjugate points. In order to obtain a successful matching between an image and vector data, the datasets must contain polygonal features like forest, villages, grassland or lakes. This approach will fail when polygonal features can not be found, like in the high resolution urban areas. Flavie and Fortier [10] tried to find the junction points of all detected lines, than matched the extremities of the road segments with the image junctions. Their method suffers from the high computation cost of finding all possible junctions of detected lines on images. Another approach, which utilizes the road axes detected from vector data to verify the extracted line segments to determine where the roads are, was proposed in [4]. This approach uses the vector data knowledge only for checking the extracted lines, thus it also takes a long time to detect road segments.

Our proposed conflation approach takes the knowledge, such as online data sources or road segment direction and road intersections provided by vector data to alleviate the problems of finding control points from aerial images. Therefore, we can efficiently acquire control points on imagery. VMF filter is utilized to remove inaccurate control point pairs to obtain better alignments. The VMF filter uses the fact that the control points on the vector data and the counterparts on the imagery are related by similar transformations in a small region.

7. Conclusion and Future Work

The main contribution of this paper is the design and implementation of a novel information integration approach to automatically annotate and integrate spatial data-

sets. Our approach utilizes the online data sources and intersection points detected by localized image processing as control points. Moreover, the inaccurate control points are removed by our proposed filter. Experimental results on the city of El Segundo and the city of Adams Morgan demonstrate that our approach can accurately align and annotate satellite images with vector data.

We plan to further improve the integration result by an iterative conflation process. The process could work as follows: the vector-image conflation operations, automatic control point pairs generation and vector to imagery alignment, are alternately applied until no further control point pairs are identifiable. We also intend to extend our approach in several ways. Extending our approach to integrate multiple satellite images of the same area is one possible topic. Our approaches can be utilized to align both spatial image datasets with some other spatial vector dataset. Another possible topic is extending the localized image processing technique to improve the performance of the Building Finder application. Although the Building Finder application has successfully integrated information from various geo-spatial data sources to locate the buildings in the imagery, the boundaries of the buildings are represented by rectangles instead of the exact building edges. Utilizing the localized image processing within each rectangle to further refine the building boundaries is a promising future research direction.

Acknowledgements

We would like to thank Dr. Jose-Luis Ambite for his comments on various aspects of this project. We would also like to thank Bo Han for his help with online control point identification. This material is based upon work supported in part by the Defense Advanced Research Projects Agency (DARPA) and Air Force Research Laboratory under contract/agreement numbers F30602-01-C-0197 and F30602-00-1-0504, in part by the Air Force Office of Scientific Research under grant numbers F49620-01-1-0053 and F49620-02-1-0270, in part by the United States Air Force under contract number F49620-01-C-0042, in part by the Integrated Media Systems Center, a National Science Foundation Engineering Research Center, under cooperative agreement number EEC-9529152, and in part by a gift from the Microsoft Corporation.

References

1. J. Astola, P. Haavisto, and Y. Neuvo. *Vector Median Filter*. *In Proceedings of IEEE*. 1990.
2. T. Barclay, J. Gray, E. Strand, S. Ekblad, and J. Richter, *TerraService.NET: An Introduction to Web Services*, Microsoft Corporation.
3. T. Barclay, J. Gray, and D. Stuz, *Microsoft TerraServer: A Spatial Data Warehouse*. 1999, Microsoft Corporation.
4. A. Baumgartner, C. Steger, C.Wiedemann, H. Mayer, W. Eckstein, and H. Ebner., *Update of Roads in GIS from Aerial Imagery: Verification and Multi-Resolution Extraction.* IAPRS, 1996. **XXXI**.
5. M.d. Berg, M.v. Kreveld, M. Overmars, and O. Schwarzkopf, *Computational Geometry: Algorithms and Applications*. Springer-Verlag, 1997.
6. ESEA, Inc., Map Merger: Automated conflation tool for ArcGIS, http://www.conflation.com/map_merge/ 2002

7. S. Filin and Y. Doytsher. *A Linear Conflation Approach for the Integration of Photogrammetric Information and GIS Data*. *IAPRS*. 2000. Amsterdam.
8. M.A. Fischler and R.C. Bolles, *Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography.* Communications of the ACM, 1981. **24**.
9. M.A. Fischler, J.M. Tenenbaum, and H.C. Wolf, *Detection of Roads and Linear Structures in Low Resolution Aerial Images Using Multi-Source Knowledge Integration Techniques.* ComputerGraphics and Image Processing, 1981. **15**(3): p. 201-223.
10. M. Flavie, A. Fortier, D. Ziou, C. Armenakis, and S. Wang. *Automated Updating of Road Information from Aerial Images*. *American Society Photogrammetry and Remote Sensing Conference*. 2000.
11. A. Fortier, D. Ziou, C. Armenakis, and S. Wang, *Survey of Work on Road Extraction in Aerial and Satellite Images, Technical Report*. 1999.
12. F. Harvey and F. Vauglin. *Geometric Match Processing: Applying Multiple Tolerances*. *Proceedings of International Symposium on Spatial Data Handling (SDH)*. 1996.
13. C. Heipke, H. Mayer, and C. Wiedemann, *Evaluation of Automatic Road Extraction.* IAPRS, International Society for Photogrammetry and Remote Sensing, 1997. **32**(3-2(W3)).
14. H. Hild and D. Fritsch, *Integration of vector data and satellite imagery for geocoding.* IAPRS, 1998. **32**.
15. J.-R. Hwang, J.-H. Oh, and K.-J. Li. *Query Transformation Method by Delaunary Triangulation for Multi-Source Distributed Spatial Database Systems*. *ACMGIS*. 2001.
16. J.S. Liadis, *GPS TIGER Accuracy Analysis Tools (GTAAT) Evaluation and Test Results*. 2000, TIGER Operation Branch, Geography Division.
17. I. Muslea, S. Minton, and C.A. Knoblock, *Hierarchical Wrapper Induction for Semistructured Information Sources.* Autonomous Agents and Multi-Agent Systems, 2001. **4**(1/2).
18. R. Nevatia and K.R. Babu, *Linear Feature Extraction and Description.* Computer Graphics and Image Processing, 1980. **13**: p. 257-269.
19. R. Nevatia and K. Price, *Automatic and Interactive Modeling of Buildings in Urban Environments from Aerial Images* . IEEE ICIP 2002, 2002. **III**: p. 525-528.
20. D. Nicklas, M. Grobmann, S. Thomas, S. Volz, and B. Mitschang. *A Model-Based, Open Architecture for Mobile, Spatially Aware Applications. International Symposium on Spatial and Temporal Databases*. 2001. Redondo Beach, CA.
21. K. Price, *Road Grid Extraction and Verification.* IAPRS, 1999. **32 Part 3-2W5**: p. 101-106.
22. A. Saalfeld, *Conflation: Automated Map Compilation*, in *Computer Vision Laboratory, Center for Automation Research*. 1993, University of Maryland.
23. W. Shi and C. Zhu, *The line segment match method for extracting road network from high-resolution satellite images.* GeoRS, 2002. **40**(2).
24. C. Steger, H. Mayer, and B. Radig. *The Role of Grouping for Road Extraction*. *Automatic Extraction of Man-Made Objects from Aerial and Space Images (II)*. 1997. Basel, Switzerland.
25. S. Tejada, C.A. Knoblock, and S. Minton, *Learning Object Identification Rules for Information Integration.* Information Systems, 2001. **26**(8).
26. U.S.Census Bureau - TIGER/Lines, http://www.census.gov/geo/www/tiger/ 2002
27. V. Walter and D. Fritsch, *Matching Spatial Data Sets: a Statistical Approach.* International Journal of Geographic Information Sciences, 1999. **5**.
28. M.S. White and P. Griffin, *Piecewise Linear Rubber-Sheet Map Transformation.* The American Cartographer, 1985. **12**(2): p. 123-131.
29. S. Yuan and C. Tao. *Development of Conflation Components. Proceedings of Geoinformatics*. 1999.