

# Georeferencing Maps With Contours

Peter BAJCSY

National Center for Supercomputing Applications  
605 East Springfield Avenue, Champaign, IL 61820

and

Tyler J. ALUMBAUGH

National Center for Supercomputing Applications  
605 East Springfield Avenue, Champaign, IL 61820

## ABSTRACT

In this paper, we present an overview of the georeferencing capabilities that enable registration of digital maps (raster data) with contours (vector data). This paper outlines the theory and standards used for proper geographic referencing of digital maps, and focuses on extraction of georeferencing information from TIFF and ESRI IMG raster file formats. Given 2D maps and 3D contours, georeferencing transformations are described for translating between spatial points specified in three different coordinate systems: latitude/longitude, UTM, and pixel. We interpret and utilize georeferencing parameter information embedded in different types of digital maps to provide a system that can geo-register contours for several popular map types and their file formats.

**Keywords:** Geographical Information Systems, Georeferencing

## 1. INTRODUCTION

*Georeferencing*, or geographic referencing, is the name given to the process of assigning values of latitude and longitude to features on a map. Latitude (lat) and longitude (long) describe points in three-dimensional (3D) space, while maps are inherently two-dimensional (2D) representations. With the advent of computers, modern maps are usually stored as digital images since they represent raster information similar to 2D image information. The steps involved in georeferencing a digital map image can vary among map image specification types, but the end result is the ability to retrieve the lat/long coordinates for any point on the georeferenced map. This capability is useful because lat/long coordinates precisely define the position of an object on the Earth.

The reader will note that the georeferencing process involves *transformations* that occur for a variety of reasons. Some transformations are related to different file formats, others to different metadata representations, some to resolution, and others to the translations between various 2D and 3D coordinate systems. The translation between coordinate systems is the only transformation dictated by the problem domain. The others, while very important, are reflections of the lack of a single standard for specifying data in the domain.

Lat/long coordinates define a point on a 3D model of the Earth, while map coordinates represent a pixel – a row and column location on a 2D grid obtained from projecting some 3D model of the Earth onto a plane. 2D maps are easy to display and facilitate distance measurement, while 3D coordinates are accurate but cumbersome and have no standard length for

different degrees of latitude and longitude. The best way to define the position of an object on a map is with relative horizontal (column) and vertical (row) distances: “Three kilometers north of object A and two kilometers west of object B”. In contrast, the best way to specify the position on a 3D sphere is with relative angular offsets: “Five degrees north and six degrees west of A.”

In general, georeferencing involves transformations of 3D (lat/long) coordinates to and from 2D (map) coordinates. Georeferencing transformations must reflect the geometric properties of the earth model and projection plane, and also support a mechanism for expressing distances and relative positional information across different coordinate systems.

Given 2D and 3D coordinate systems, there are three possible classes of transformations. First, those that go from 2D-to-2D, bringing into alignment points that represent identical real-world locations. An example of this is multi-modal raster data fusion, for instance, the fusing of synthetic aperture radar (SAR) images with electro-optical (EO) or infrared (IR) images. The second class of transformations, 3D-to-3D, is used for information integration such as the synthesis of road network and river information. The third class of transformations operates between 2D and 3D coordinates. Identifying the location of water wells defined in 3D lat/long coordinates on a 2D digital terrain map is an application example in this class of transformations. The transformations across coordinate systems of different dimensions, where one of the coordinate systems represents the Earth, are also called geo-registration. In our work, we focus on the third class of transformations, those that support georeferencing maps (2-D raster data) with contours or boundaries of regions (3-D vector data).

Fields that apply georeferencing include geology, hydrology, atmospheric science, archeology, earthquake engineering, forestry, environmental engineering, water quality research, ecological research, military operations and territorial insurance. For example, the petroleum industry needs to know precise drilling locations, so accurate georeferenced maps are very important. The United States Geological Survey (USGS) and the Defense Mapping Agency both create and make extensive use of georeferenced images, many of which are available to the private sector.

Users of georeferencing information frequently exchange data and therefore have adopted some georeferencing terminology and Earth modeling standards. However, not all georeferenced maps follow the same standards, so identical information content can be represented using different data formats or map

types. The developed software tools are designed to work seamlessly with several of the most popular standards and are part of the Image To Knowledge (I2K) software [7].

In this paper, we outline the theory and standards used for proper geographic referencing of digital images and focus on extracting georeferencing information from TIFF and ESRI IMG raster file formats. In Section 2 we describe other GIS software packages and some basic georeferencing principles. Individual georeferencing transformations and sources of necessary georeferencing information are presented in Sections 3 and 4. Section 5 covers specific details of georeferencing in I2K and Section 6 presents a summary and discussion of future work.

## 2. CURRENT SOLUTIONS

A software package that can accept geospecific data and provide geographic referencing is called a *Geographic Information System* (GIS). One example of a GIS system is ArcGIS from ESRI [2], with components that include ArcView and ArcExplorer. ENVI, the Environment for Visualizing Images from Research Systems, Inc. [6], and a suite of applications known as the USGS Mapping Science Software from the U.S. Geological Survey [3] are other GIS systems.

There is nearly limitless information available describing georeferencing (see [1], [12]), but the level of understanding necessary to correctly georeference a single image can be rather daunting. The difficulty is due in part to the complications of projecting a three-dimensional surface, the Earth, onto a two dimensional object, a map. Understanding something about this process goes a long way towards understanding why certain parameters are needed to geographically orient a digital image.

All GIS systems require a certain amount of metadata about an image before it can be properly georeferenced. This metadata, sometimes referred to as the *map parameters*, includes information such as the projection of the map. The projection conveys the manner in which the three dimensional features of the Earth have been distorted onto a two dimensional image. There are hundreds of map projections in use, and the way the distortion occurs influences the usefulness of the map in different domains. A list of nearly all map projections in use today can be found in [12]. Fortunately, organizations like the USGS and the Defense Mapping Agency use only a small subset of these.

Another map parameter used for georeferencing is the approximation of the Earth used in the projection. The Earth is not a perfect sphere, so it is first modeled as some ellipsoid, possibly a sphere, before being projected onto a plane. Other types of metadata are the *projection center* (where the distortion of the map is minimal) and the *insertion point* (a point on the image used to extrapolate data about other points on the image). The formulas used to do the georeferencing depend upon additional map parameters, increasing the amount of prerequisite knowledge.

The developed software can perform all the basic georeferencing transformations provided the necessary metadata has been specified correctly. The presented georeferencing functionality is similar to other GIS software packages, and works with a range of underlying map types. The georeferencing feature in I2K was motivated by the need to enable statistical GIS data processing that is not available in

other GIS software packages, for example, statistical measures of raster data (maps) over territories defined by vector data. The functionality of the developed software has been tested thoroughly with standard examples from the literature [14], and the tests are described in the Section 5 of this document.

## 3. GEOREFERENCING TRANSFORMATIONS

An interface has been developed to provide uniform georeferencing capabilities across a range of map types by abstracting away the specific parameters of different map types and maintaining a uniform and consistent internal representation. This design allows for the addition of as yet unsupported map types through the development of new code modules. In I2K, a *GeoImageObject* holds all georeferencing information. The fields with geographic referencing metadata in the *GeoImageObject* are populated when a new image is loaded, and dictate the algorithms applied during the transformation process.

The georeferencing system is built around coordinate transformations to and from three types of location representations: 2D map pixels (column and row), 3D lat/long coordinates, and 2D Universal Transverse Mercator (UTM) values. These transformations are shown in Figure 3-1.

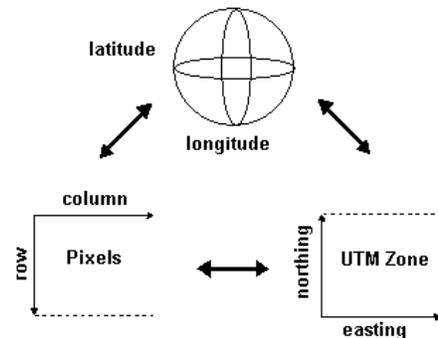


Figure 3-1: Coordinate transformations supported by I2K

### I2K Model Types

As mentioned earlier, a *GeoImageObject* holds all georeferencing information in I2K and the fields of the object are populated with geographic referencing information when a new image is loaded. In order to perform the desired georeferencing transformations, a *model type* (the I2K equivalent of a *datum*, for the GIS savvy) must be set based on the map parameters that are loaded. The model type is not set if all the required metadata is not found, and georeferencing cannot be performed without the model type.

UTM coordinates are used in maps with many different kinds of projections, but also lend their name to a number of them. Each projection type, along with the various parameters needed to perform the transformations, make up a model type. Each model type has specific, private methods that can be called to perform the six transformations between pixel, UTM, and lat/long coordinates. The model types currently supported are:

1. The UTM Northern Hemisphere.

2. The UTM projection called the Ordnance Survey of Great Britain from 1936, OSGB 1936.
3. The Lambert Azimuthal Equal Area map projection.

The OSGB 1936 model type has been used primarily for testing purposes. The Lambert Azimuthal Equal Area Projection is, in general, only used for large-scale maps.

### Transformations between Latitude / Longitude and UTM Northing / Easting

The UTM system is two-dimensional and the latitude/longitude system is three-dimensional, so making transformations between the two is quite complex. The two methods that perform transformations between UTM and lat/long are based on equations in [5], [12], [13] and [14].

The transformation from UTM to lat/long is referred to as *Transformation 1*, and the transformation from lat/long to UTM is referred to as *Transformation 2*. Given the relative obscurity of the UTM system, one might question the purpose behind these transformations. The answer lies in the fact that many standard georeferencing algorithms (such as those that follow) produce results in UTM values. The algorithms underlying these methods are chosen based on the metadata loaded into the GeoImageObject in conjunction with the image.

The model type determines whether the standard Molodensky equations ([5], [13]) or the Lambert projection equations [14] are used to perform these two transformations. The Molodensky and Lambert equations require parameters for both the three-dimensional and two-dimensional coordinate systems. We present what must be known about the three-dimensional object and about the two-dimensional plane in the following subsections.

**Ellipsoid Parameters for Latitude/Longitude Transformations:** Since the Earth is not a sphere, an ellipsoid of some type is used as a model for the Earth. Only for very large-scale maps, such as the Lambert Azimuthal Equal Area projection, will one use an actual sphere to model the Earth. In all other cases, a special ellipsoid of some type is specified. These geodetic ellipsoids used as models should always be referred to as *oblate ellipsoids of revolution*, but are referred to simply as *ellipsoids* in all georeferencing literature. All geodetic ellipsoids can be specified by two quantities: a semi-major axis and a semi-minor axis. The equations used in Transformations 1 and 2 make use of two related values to define an ellipsoid: the semi-major axis and the square of the eccentricity of the ellipsoid. The *eccentricity* value, denoted as  $e$ , is a measure of the amount by which an ellipse varies from a circle. The square of the eccentricity is calculated with this equation:

$$e^2 = 2f - f^2 \quad (1)$$

The *flattening* of the ellipse,  $f$ , is defined by:

$$f = (a - b)/a \quad (2)$$

In the flattening equation [Eq.(2)],  $a$  is the ellipsoid semi-major axis and  $b$  is the ellipsoid semi-minor axis. Often  $(a - b)$  is quite small compared to  $a$ , and it is common to provide the flattening quantity as  $1/f$ , or the *inverse flattening*.

**Two-Dimensional UTM Map Parameters:** In general, UTM map parameters should include information pertinent to a map's 2D coordinate system and information about the 3D to 2D projection used to obtain the map. Ideally,

the map parameters will provide at least the following information: projection type, projection center, false easting, false northing, UTM zone (if applicable), vertical resolution, horizontal resolution, and the tie point.

Of all map parameters, the map projection is perhaps the most important because it determines which set of equations is used to perform the transformations. Most of the other map parameters are used as inputs to that particular set of equations. The map projection is generally subsumed into the model type described in Section 3.1. Multiple model types might have the same map projection, but other map parameters could be different.

The map resulting from a projection will not have the same amount of distortion at each point. Different parts of the map plane are oriented to the Earth model differently, resulting in varying degrees of distortion. Generally, the distortion on a map will be minimal at its *projection center*. It is important to note that the projection center may or may not be in the viewable area of an image.

Two other quantities that are specific to a projection are the *false easting* and *false northing*. In a UTM map projection, points to the west or south of the projection center can have negative UTM values. In order to avoid the inconvenience introduced by negative values, false easting and false northing values are often added to computed northings and eastings to keep all values positive.

### UTM Northing/Easting to and from Pixel Column and Row

It is also useful to go from UTM northing/easting to pixel-valued column and row (Transformation 3) and back (Transformation 4). Since these are both horizontal/vertical 2-D coordinate systems, the transformations are linear equations mapping from one Cartesian system to another.

The following fields must be populated for Transformations 3 and 4 to produce correct output: (1) a tie point/ insertion point, (2) resolution/scale, (3) parameters and units of measure. A *tie point*, also called an *insertion point*, is a pair of coordinates  $[(i,j), (x,y)]$  that tie together a pixel value  $(i,j)$  and its UTM equivalent  $(x,y)$ .

### Latitude/Longitude to and from Pixel Column and Row

The most frequent application of georeferencing is to translate from a pixel value, that is, a column and row of an image, to a latitude/longitude pair. Transformation 5 performs this calculation, while Transformation 6 does the reverse calculation from a latitude/longitude pair to an image column and row. These latitude/longitude  $\leftrightarrow$  column and row transformations can be accomplished via calls to Transformations 1 through 4 discussed above. Specifically, Transformation 5 is carried out by a call to Transformation 4 followed by a call to Transformation 1. Calling Transformation 2 and then calling Transformation 3 accomplishes Transformation 6.

## 4. SOURCES OF GEOREFERENCING INFORMATION

In order to perform the geographic referencing transformations correctly, a number of map parameters must be given along with the map data. While some file formats were designed to contain georeferencing information, for example the ERDAS

IMG file format, the file types of most digital maps were not designed to accommodate this metadata. For these file formats, of which the Tagged Image File Format (TIFF) is one example, the metadata must be provided in some manner for the map to be of any use. Although a number of approaches have been taken, the methods for including the metadata have not yet reached a point of standardization that enables a novice to quickly find the necessary information.

In this section we describe how geographic referencing information can be extracted from TIFF and IMG files, and detail the georeferencing information sources for the TIFF and IMG file formats.

### TIFF Files

The georeferencing information extraction from TIFF world files (tfw) and/or private tags is based on the literature available in [10], [11]. The current solution for the TIFF format uses a combination of information extraction approaches that are based on three variants of TIFF files containing georeferencing information. Assuming that sufficient data is provided in some combination of the three sources, we can use the georeferencing interface described earlier. The three TIFF file variants are:

- One or more standardized files are distributed along with TIFF image data as .tfw and/or .txt files.
- The metadata is encoded in the image file using private TIFF tags.
- An extension of the TIFF format called *GeoTIFF* is used.

Given multiple information sources, it is possible to read conflicting values for the same geographic field. For example, tie points and resolution values can be specified in both the TIFF world file and the private TIFF tags. When both a tfw file and the private tags are present, the tfw values have priority.

Some georeferencing information may be specified either in deg/min/sec or UTM meters. In particular, if the information in private tag 33922 is specified in deg/min/sec, the developed software can perform the necessary transformation.

Finally, in the absence of necessary information, some transformations will not be executed. For example, if both the tfw file and private tags 33550 and 33922 are missing, Transformations 3 and 4 cannot be performed.

**TIFF World Files:** Geospecific information for a map in the TIFF format is often included in separate file with the same root file name, but with the extension '.tfw' [10]. A tfw file is a short ASCII file that contains six values, four of those being the values discussed in Section 3.3.

Value	Description
30.000000	Column (horizontal) resolution
0.000000	Rotation in the horizontal direction
0.000000	Rotation in the vertical direction
-30.000000	Row (vertical) resolution
250000.00	Easting value of tie point

650000.00	Northing value of tie point
-----------	-----------------------------

**Table 4-1: Sample TIFF world file values**

Table 4-1 shows typical values and descriptions for a TIFF world file. In looking at the tfw values, observe that the easting and northing values are available directly from the file. The horizontal and vertical resolution factors are also in the file, and in this example the vertical resolution is a negative value. A tfw file implies that the  $(i,j)$  value of the tie point is either the upper left corner or the lower left corner of the image. If the vertical resolution is given as a negative value, then the tie point of the image is implied to be the upper left, that is (0,0). If the vertical resolution is given as a positive value, then the tie point of the image is implied to be the lower left, that is  $(0, \text{number\_of\_rows}-1)$ , where *number\_of\_rows* is the number of rows in the image.

**Private TIFF Tags:** The TIFF file format uses a data structure known as a *tag* to supply information about an image. Every tag within the TIFF specification has a particular number associated with it, called its *value*. Any software that reads a TIFF file reads a tag structure by first examining the tag value, and then interpreting the meaning of the tag based on the tag value.

The TIFF specification [7] defines the values and meanings of every valid tag, so no single entity can create a TIFF tag of arbitrary value for its exclusive use. However, it is possible for a company to register one or more TIFF tags for private use, and a number of these so-called private tags are quite useful for geographic referencing [11]. Four private tags are supported in I2K.

- **Private Tag 33550**

Private tag 33550 is referred to as the *ModelPixelScaleTag* and consists of three IEEE double precision floating-point numbers (Horizontal (x) resolution, Vertical (y) resolution and Elevation (z) resolution). This structure has built-in support for digital elevation maps.

- **Private Tag 33922**

Private tag 33922 is referred to as the *ModelTiePointTag*. It usually consists of six IEEE double precision floating-point numbers organized as shown in the following table. Tag 33922 may contain a sequence of  $[(I,J,K), (X,Y,Z)]$  values, specifying multiple tie points (Column number of tie point in pixel space, Row number of tie point in pixel space, Elevation value of pixel  $(I,J)$ , Easting value in model space, Northing value in model space, Elevation of  $(X,Y)$  in model space).

- **Private Tag 34735**

Private tag 34735 is called the *GeoKeyDirectoryTag*. This tag is used to implement the GeoKey data structure for GeoTIFF files, an extension of TIFF.

- **Private Tag 34737**

Private tag 34737 is referred to as the *GeoAsciiParamsTag*. This tag holds the ASCII values of any GeoKeys that contain ASCII characters. An example is GeoKey 3073.

**GeoTIFF Keys:** The private TIFF tag solution worked well for the individual companies who registered the tags. However, for third-party GIS companies, the solution was

not ideal because their software had to interoperate with each company's standards. The third-party companies wanted a public standard that would be platform independent and available to anyone. The GeoTIFF format is an open standard built from the TIFF 6.0 specification. Its purpose is to include geographic information in a TIFF document in the form of GeoTIFF keys.

GeoTIFF keys, commonly known as *GeoKeys*, can be described as data structures built on top of the tag structure. A GIS reader that does not support the GeoKey specification reads all GeoKeys simply as TIFF tags it does not understand. Following the example of the TIFF tag, each GeoKey has a unique value called an *ID number*. By efficiently using the TIFF tag structure, the GeoTIFF standard allows for GeoKeys with ID numbers ranging from 0 to 65,535 while using a maximum of six registered tags.

- **GeoKey 1024**

GeoKey 1024 is called the *GTModelTypeGeoKey*. It consists of one short (2-byte unsigned integer) value that specifies the general type of coordinate system used. Geographic and geocentric coordinate system images are less common than images with a projection coordinate system.

GeoKey 1024	
Possible Value	Description
1	A projection coordinate system is being used
2	A geographic coordinate system is being used
3	A geocentric coordinate system is being used

- **GeoKey 1026**

GeoKey 1026 is the *GTRasterTypeGeoKey*. The purpose of this key is to remove the ambiguity inherent in representing a coordinate system with raster data. The key contains one short (2-byte unsigned integer) with two possible values that indicate how to interpret a raster coordinate (a pixel).

GeoKey 1024	
Possible Value	Description
1	Raster pixel represents area
2	Raster pixel represents point

As an example, consider a raster coordinate of (1,1). If the value of GeoKey 1026 is 1, it indicates that the raster coordinate (1,1) refers to the entire square grid with boundaries (0,0) in the upper left and (1,1) in the lower right. More generally, in this system a coordinate (*i,j*) refers to the 1 unit square area with (*i,j*) as its lower right corner. In contrast, if the value of GeoKey 1026 is 2, a pixel value refers only to point indicated by the intersection in pixel space of a vertical line *i* units to the right of

the origin with a horizontal line *j* units down from the origin. In our example for raster coordinate (1,1), this would be the point 1 unit right from the origin and 1 unit down. For details see Section 2.5.2.2 in [8].

- **GeoKey 3072**

GeoKey 3072 is the *ProjectedCSTypeGeoKey*. If GeoKey 1024 has value 1, then this key indicates the specific projection that was used. There are 12,761 allowable values for this key, ranging from 20000 to 32760, with each one a well-defined projection on a particular geographic location. See Section 6.3.3.1 in [8] for details.

- **GeoKey 3073**

The *PCSCitationGeoKey*, GeoKey 3073, exists to compensate for the fact that GeoKey 3072 is not a user-friendly way to specify a map projection. The value of GeoKey 3073 is an ASCII string intended to give the user understandable information about the map projection. Often it consists of just a few words providing information such as the UTM zone, the ellipsoid used, or other geographic metadata about the image. The string is shown to the user with other geographic information in the *GeoInfo* window in I2K (see Section 5).

### ERDAS IMG Files

I2K is also capable of georeferencing maps stored in the ERDAS IMG format. For these files, the metadata is always included in a separate header file. The header file has the same root file name as the image, but with '.hdr' as its extension. Many types of metadata about the img file are included in the header file in a 'keyword = value' format, with the important keyword for geographic referencing being *map info*.

The map info entry in the header file has the format:

```
map info = { projection_name, i, j, x, y, horizontal_resolution,
            vertical_resolution
            UTM_zone, 'North' or 'South' }
```

The individual values in the map info entry are:

*projection\_name*: The type of projection used in the image, often UTM.

$[(i,j),(x,y)]$ : The tie point.

*horizontal\_resolution*: The horizontal resolution of a pixel.

*vertical\_resolution*: The vertical resolution of a pixel.

The last two individual values are only present if the projection specified is the UTM Northern Hemisphere or the UTM Southern Hemisphere projection:

*UTM\_zone*: The UTM zone in which the region is located.

'North' or 'South': Indicates if the image is in the Northern or Southern Hemisphere of the specified zone.

## 5. GEOREFERENCING EXAMPLES

The developed software enables two primary functions related to georeferencing files. First, it allows a user to view

georeferencing information for any loaded map. Second, it uses georeferencing to register raster files, such as maps, and vector files, such as boundaries or contours, stored in the shape file format.

Once a georeferenced image is loaded into the image window, a user can view all of its available geospecific information by right clicking on the image and selecting *GeoInfo* from the menu. The values displayed will normally include the scale/resolution of the image, the projection type of the map, and/or the ellipsoid name from which this map was projected. An example is illustrated in Figure 5-1.

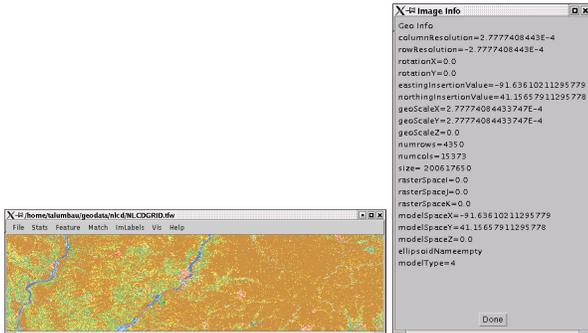


Figure 5-1: A land cover map image and its GeoInfo

A few tools in the I2K software use georeferencing functionality, for example, the *GeoFeature* tool. With this option, a user can choose to register and overlay a contour or boundary stored in a shape file onto the georeferenced image. If enough image metadata are correctly loaded to support all six transformations, the shape file with points specified by latitude and longitude will be correctly georeferenced over the image. Once the geo-registration has been completed, statistics of image attributes over each boundary can be computed. An example of geo-registration is shown in Figure 5-2.

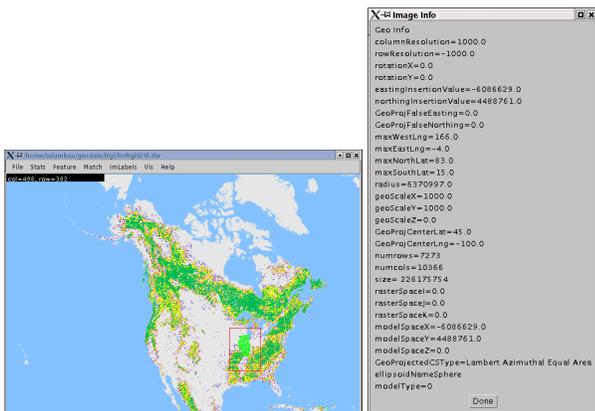


Figure 5-2: A TIFF image with forest cover labels (left) and its georeferencing information (right) can be geo-registered with a shape file containing boundaries of Illinois counties (left image – Illinois falls within indicated rectangle).

## 6. SUMMARY

The developed software provides georeferencing for several of the most common types of digital map data. The implementation allows new projection and file types to be supported with minimal additional coding effort. Considerable

effort was spent on effectively reading and utilizing the available map metadata.

To keep the software as easy to use as possible, the described extraction of georeferencing information attempts to find the required data in several places before giving up, and occasionally makes a 'best guess' if the necessary data cannot be found. The result of our work is the ability to load georeferenced images with the geospecific information embedded in the image files, included in standard additional files, or some combination of these two for the TIFF and ESRI IMG file formats.

A possible future direction for geographic referencing in I2K would be the addition of an Expert Mode. If the user knows required information about a map image whose metadata is in a different format than I2K expects, the user could simply use a GUI tool to tell I2K the necessary information. This GUI tool could take a 'wizard' format, prompting the user with a series of questions to facilitate the data entry. The direct-entry feature would require more user knowledge and expertise, but would provide immediate accessibility to the georeferencing capabilities of I2K for images in formats not yet coded into the system.

## REFERENCES

- [1] Gateway site to entire topic: <http://www.remotesensing.org>
- [2] ESRI website: <http://www.esri.com/>
- [3] USGS Mapping Science Software: <http://mapping.usgs.gov/www/products/software.html>
- [4] Educational site: <http://www.colorado.edu/geography/gcraft/notes/datum/datum.html>
- [5] Technical Information: [http://www.posc.org/Epicentre.2\\_2/DataModel/ExamplesofUsage/eu\\_cs.html](http://www.posc.org/Epicentre.2_2/DataModel/ExamplesofUsage/eu_cs.html)
- [6] Research Systems, Inc, The Environment of Visualizing Images (ENVI), <http://www.rsinc.com/envi/index.cfm>
- [7] P. Bajcsy, "Image To Knowledge (I2K)," Software Overview and Documentation, <http://www.ncsa.uiuc.edu/Divisions/DMV/ALG/activities/projects/i2k/documentation/index.html>.
- [8] TIFF Revision 6.0. Aldus Corporation. Copyright 1986-88, 1992.
- [9] GeoTIFF Revision 1.0. Ritter, Niles. Ruth, Michael. 2000.
- [10] *Map Projections – A Reference Manual*. L. M. Bugayevsky and J. P. Snyder. Taylor and Francis,
- [11] Information on tfw files: <http://www.genaware.com/html/support/faqs/imagis/imagis15.htm>
- [12] Information on private TIFF tags: <http://remotesensing.org/geotiff/spec/geotiff2.6.html>
- [13] *Understanding Map Projections*. Kennedy, Melita. Kopp, Steve. Environmental Systems Research Institute, Inc.
- [14] Molodensky equations reference [http://www.posc.org/Epicentre.2\\_2/DataModel/ExamplesofUsage/eu\\_cs34h.html](http://www.posc.org/Epicentre.2_2/DataModel/ExamplesofUsage/eu_cs34h.html)
- [15] Lambert projection equations <http://mathworld.wolfram.com/LambertAzimuthalEqualAreaProjection.html>
- [16] Online reference to transformation equations [http://www.posc.org/Epicentre.2\\_2/DataModel/ExamplesofUsage/eu\\_cs.html](http://www.posc.org/Epicentre.2_2/DataModel/ExamplesofUsage/eu_cs.html)
- [17] T. Alumbaugh and P. Bajcsy, "Georeferencing Maps with Contours in I2K," Technical Report NCSA-ALG-02-0001, October 2000.