Supervised Change Detection on Simulated Data employing Support Vector Machines

Christodoulos PSALTIS, Charalabos IOANNIDIS, Greece

Key words: Photogrammetry, Machine learning, Change detection, Support Vector Machines

SUMMARY

The increasing need for easy and cost effective updating of Geographic Information Systems (GIS), the wide availability of inexpensive high resolution data and the exponential increase in computing power, fuel extensive research in automatic methods for change detection of manmade objects. This paper illustrates a methodology to accomplish such tasks. The main idea of the proposed procedure follows the supervised classification paradigm. The first step is to layer the available data for the same region in different time periods. Then evaluate a number of predefined cues for the whole region and use some manually collected positive and negative samples to train a classifier. Finally this classifier can be used to assert change in the remaining data. The base data chosen are very high resolution orthoimages and digital surface models (DSMs) because they offer both the radiometric and geometric information needed for robust change detection. The classifier selected is the support vector machines (SVM) algorithm because it offers some significant advantages over alternative methods. These advantages include convergence to a global maximum, requirement of a small number of training samples and the availability of good open source implementations. In the paper, emphasis is given in testing the proposed strategy with simulated data, to access its validity and performance aspects. The simulated data were produced automatically with a program developed especially for this purpose. Noise is gradually imported to the testing data to make them more realistic. Noise can be a combination of random radiometric noise for the images, geometric noise for the buildings depicted in the images and finally height noise for the DSMs. Different setups were planned and implemented in all of which the results indicate that the proposed methodology has good performance.

Supervised Change Detection on Simulated Data employing Support Vector Machines

Christodoulos PSALTIS, Charalabos IOANNIDIS

1. INTRODUCTION

Automatic change detection in the field of photogrammetry, deals with the dynamic phenomenon of land use change at large data scales. Usually the objects which are monitored are manmade structures like roads and buildings. This process is necessary for map updating which is an otherwise expensive and time consuming task.

Although in its general form the problem of automatic change detection is complex and difficult to solve, on well defined applications it is possible to achieve good results by imposing certain constraints. Depending on the special needs of an application these constraints may vary, thus the available change detection methods in international bibliography are many and diverse. However the basic attributes of all these algorithms are mainly common and the presented diversity results from different combinations of attribute values. Thus it is best to describe these attributes and how they impact the final method rather than overview groups of methods. This analysis is also helpful for designing new methods and approaches by making certain decisions in each attribute, based on the application needs.

The following are roughly the most distinctive attributes:

- Scale of the changes to be detected. This parameter is application depended and greatly influences choices in following attributes. Scale of change can be illustrated with a simple example. If the user needs to detect changes in the development of the urban area limits (Hofmann et al. 2006) then whole groups of buildings can be thought as one and thus the data needed to depict this type of change can have small scale. If the user in interested in detecting change in single building level, then the data necessary should be of large scale in order to better represent the objects of interest and thus this is considered a large scale change detection problem (Moeller & Blaschke, 2006).
- The type of the basic comparison unit. Basic comparison unit is called the feature which is compared between two time periods for assertion of change. Its type expresses the information level it carries, ranging from low, i.e. grey tone values, to high level information, i.e. object classes (Straub et al., 2000). The highest the level of information the more robust the method is. At the same time however, it is more difficult to develop and maintain. The nature of the basic unit is primarily decided depending on the scale of changes to be detected. In particular large scale changes demand more sophisticated comparison units.
- The number of steps in which the process is completed. At this point it is decided if changes will be detected in one or two steps. The two step approach involves the extraction of objects in both time periods and then the comparison between them to decide what has changed (Blaschke, 2005). On the other hand it is possible to complete the same procedure in a single step by layering the two time periods in a single new product, i.e. a difference

image, and then deciding which features of the new dataset are indicating changes (Psaltis & Ioannidis, 2008). Two step approaches have the disadvantage of inserting errors in the whole process due to the two extraction phases. Errors even in one phase will lead to errors in change detection. On the other hand they are more noise resistant.

- The path to change detection and incorporation of a priori knowledge. Namely there are two basic ways to go for change detection, either bottom-up (Psaltis & Ioannidis, 2008) or top-down (Hall, 2003). Bottom-up methods are considered those which by dealing initially with raw data lead to change assertion. The opposite is true for top-down approaches where certain models of change are searched and matched to the raw data. In both cases it is necessary to use some a priori knowledge to bridge the semantic gap between data and change model. These data differ in complexity according to the needs of each method, but as previously mentioned, higher level knowledge better fits large scale change detection.
- Deterministic or stochastic approach of the problem. Modeling change can be formulated in any one of the aforementioned ways. In the first case there is a decisive answer as to what change is (Song & Li, 2007) whereas in the second there is a measure of how possible change is (Canty & Nielsen, 2004). Stochastic methods have the advantage of providing a concrete quality measure of the results, possibility, and they often are more flexible and extensible. On the other hand they are more difficult to develop and often face computational challenges.
- Level of automation. In this attribute one can discern three main approaches; autonomous, automatic and semi-automatic methods. In autonomous method the user just imports data to the algorithm and gets an output without any further customizations (Canty & Nielsen, 2004). In automatic methods, users have to manually tune a set of parameters before they get the desired output (Wang et al., 2007). In semi-automatic methods users complete a training phase inputting positive and negative samples of change before the algorithm is able to predict changes in the rest of the dataset (Mo et al., 2008). Higher levels of automation mean less effort from the users, but usually they also mean lower levels of accuracy and less noise tolerance.
- Type of data used. This selection depends mainly on the application scenario and the scale of the changes to be detected. Today there is a wide range of data type available in many scales and with different characteristics. Data can be divided in two major categories raster and vector. Raster data mainly include images from different types of sensors like airborne or spaceborne cameras, SAR sensors and thermal sensors (Zhigao et al., 2006). Vector data mainly include maps, cadastral polygons, 3D point clouds and surface models (Song & Deren, 2007). Choosing the most appropriate type of data is impossible in some cases since for certain periods and areas there might be only certain types available. For this reason it is important to take into account this parameter before developing a change detection algorithm. However, with the gradual increase in data format availability this aspect will influence less the development process.

Based on the above analysis and the existing knowledge and experience on change detection techniques, a new automated method for buildings change detection is proposed; the procedure is developed and its results derived from a broad variety of simulation data are presented in this paper.

2. PROPOSED APPROACH FOR AUTOMATIC CHANGE DETECTION

The proposed method, in accordance with the aforementioned general attributes, is designed to detect changes in large scales - single building scale. The employed data are orthoimages and digital surface models (DSMs) of the area of interest. The orthoimages offer a good starting point, since they combine reduced radiometric and geometric noise, while at the same time offer large information context. To produce the orthoimages it is necessary to have a DSM available. The DSM provides direct geometric information which can augment the detection capabilities of the method. The basic comparison unit is mid level statistics, in order to keep the process simple yet as robust as possible. Detection is completed in a single step, to avoid building extraction in every time period. This choice minimizes the development work and helps to avoid errors introduced from the two stage approach. The method is deterministic and the role of assessing the quality of the results is assigned to a human user investing minimal effort. It is a bottom-up technique, because orthoimage creation significantly reduces noise and makes the development of a complex top-down approach unnecessary. It is semi-automatic so the final results are expected to be better than the fully automatic approach. The selected classifier is the Radial Basis Function Support Vector Machine (SVM). SVMs are reported to fit well in change detection scenarios (Zhigao et al., 2006; Mo et al., 2008) and they offer a number of significant advantages described later on.

The proposed workflow of the method was organized to satisfy the above characteristics, as following:

1. Acquisition of large scale images from different time periods for an area of interest. If the DSM is going to be produced, then image stereopairs are needed. Otherwise the DSM for each time period is also acquired from an external source.
2. Interior and exterior orientation of the images in the same reference system. The DSMs should also be in this reference system.
3. Production of orthoimages for each period; to reduce geometric noise and improve image correspondences.
4. Layering of the orthoimages and the DSMs and calculation of the appropriate feature vectors.
5. Manual labeling of some positive and negative samples of changes to train the classifier, which was selected to be a Support Vector Machine (SVM).
6. Input all new data to the trained SVM in order to classify them as changed or not.
7. Manual assessment of the resulted changes.

3. SUPPORT VECTOR MACHINES

The theoretical foundations of SVMs were set in the 60’s mainly with the efforts of V. Vapnik (Vapnik and Lerner, 1963), but it was not until 1992 that they reached their modern formulation (Boser et al., 1992). Today SVMs are used in a wide range of applications from text recognition to image analysis because of their generic nature and the positive tradeoff between advantages and disadvantages (Abe, 2005).

3.1 Formalization of the linear SVM classifier

The goal of the support vector machine is to classify a given object to one of two available classes based on its characteristics. Each object in this case is viewed as an n-dimensional vector. The basic idea behind the support vector machines is to calculate an n-1 dimensional hyperplane which best separates the two object classes from a set of already labeled objects. The two sample sets can be separated with a number of hyperplanes. The best solution is the hyperplane from which the distance to the nearest feature vector on each side is maximized. New objects can then be classified depending on which side of the hyperplane they lay. This is the most basic case of support vector machines and it is called linear classifier (Figure 1).

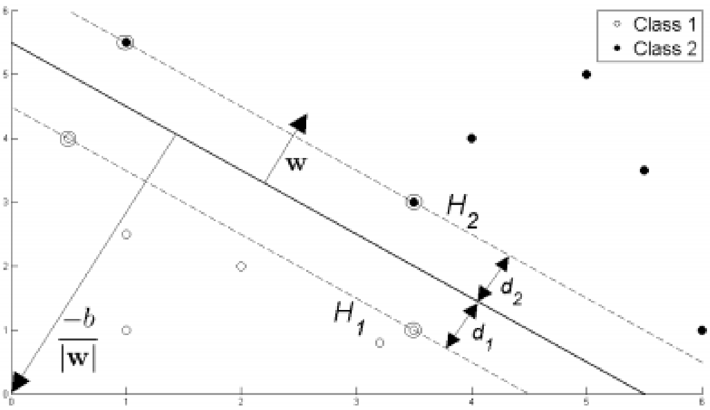

Figure 1. Linear classification example (Burges, 1998)

To formulate the above in mathematical terms suppose we have L training points, where each input xi has D attributes, dimensionality D, and belongs in one of two classes yi = -1 or +1. Our training data then are of the form {xi, yi} where i = 1…L and yi in {−1, 1}. Here it must be assumed that the data is linearly separable so that we can draw a line on a graph of x1 vs x2 separating the two classes when D = 2 and a hyperplane on graphs of x1, x2 . . . xD for when D > 2. This hyperplane can be described by **w · x + b = 0** where **w** is normal to the hyperplane and **b/||w||** is the perpendicular distance from the hyperplane to the origin. Support Vectors are the examples closest to the separating hyperplane and the aim of SVM is to orientate this hyperplane in such a way as to be as far as possible from the closest members of both classes.

In practice implementing a SVM means selecting the variables w and b so that training data can be described by:

$$xi \cdot w + b \geq +1 \text{ for } yi = +1$$
$$xi \cdot w + b \leq -1 \text{ for } yi = -1$$

These equations can be combined into: $yi(xi \cdot w + b) - 1 \geq 0$

Considering the points that lie closest to the separating hyperplane, the two planes H1 and H2 that these points lie on can be described by:

$$xi \cdot w + b = +1 \text{ for H1}$$
$$xi \cdot w + b = -1 \text{ for H2}$$

Let d1 be the distance from H1 to the hyperplane and d2 the respective distance from H2. The hyperplane's equidistance from H1 and H2 means that d1 = d2, a quantity known as the SVM's margin. In order to orientate the hyperplane to be as far from the Support Vectors as possible, we need to maximize this margin. Simple vector geometry shows that the margin is equal to $1/\|w\|$ and maximizing it is equivalent to finding:

$$\mathrm{Min}\|w\| \text{ such that } y_i(x_i \cdot w + b) - 1 \geq 0$$

3.2 Kernel functions

The above case works well when the samples are linearly separable in N-dimensional feature space, but this constraint is not always true (Figure 2). In cases where the samples are separated by a nonlinear curve there are two possible solutions; either to fit a nonlinear curve or to increase the dimensionality of feature space until they become linearly separated. The first possibility is complex and computationally exhausting whereas, due to data restrictions, you might not be able to increase the dimensionality of feature space to the desired point. To cope with this problem the proposed solution is kernel functions.

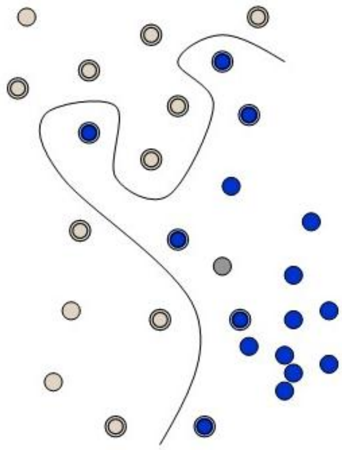

Figure 2. Non-linear hyperplane (DTREG, 2009)

Kernel functions map the data into a high dimensional space where it is possible to perform the separation. Kernel functions enable operations in the feature space without computing the coordinates of the data in that space, but rather by simply computing the inner products between the images of all pairs of data. In fact SVMs are a subclass of kernel methods.

There are almost infinite possibilities in creating a kernel function, but only a few are proved to work well in a number of situations. These include the linear kernel, the polynomial kernel and the radial basis function kernel. The linear kernel is equivalent to the linear classifier presented above. The polynomial kernel separates the samples with a polynomial curve, while the radial basis function kernel creates a separating curve around the samples. For detailed mathematical formulation of these kernels and some more refer to Christianini & Taylor (2005). To illustrate the difference between them see Figure 3.

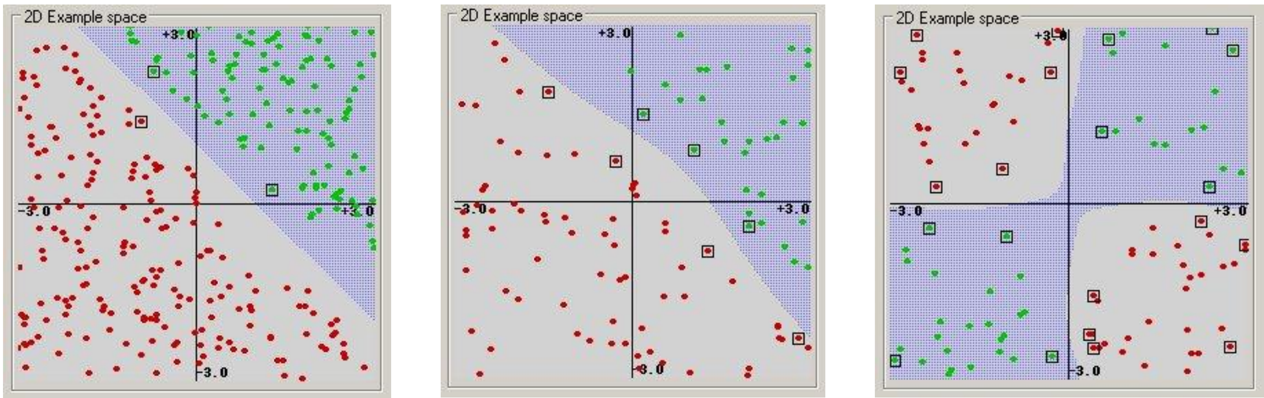

Figure 3. Left: Linear SVM classification. Middle: Polynomial kernel SVM classification. Right: Radial Basis Function SVM classification (source: DTREG, 2009)

4. SIMULATING DATA

To test the validity, effectiveness and robustness of the strategy, it was decided to simulate some datasets and employ the proposed technique to classify them. At the same time this testing can provide further information to improve the final strategy in terms of selected features, feature space dimensionality and SVM parameter tuning. Because the process of creating simulated datasets is time consuming a computer program was developed especially for this purpose. The programming language used was Python and especially the Python(x,y) distribution. Python(x, y) includes the current python release plus a set of tools and libraries especially designed for scientific computing (Raybaut, 2009).

The designed program has two basic outputs; greyscale simulations of orthoimages and simulated DSMs for the same area of interest at two different time periods. For a granular control over the final result the user has a number of available choices:

- The size of the testing area. The user sets this parameter denoting the width and height of an orthogonal area of interest in pixels.
- The area of each building in pixels. Every building in the data set will have approximately the same area. The area will not be exactly the same because of the shape parameter that follows.
- The possible shapes of buildings. Users can input a list of possible shapes their buildings might have. Before creating each building it is randomly decided what shape it will have and then based on its area it is placed on the final image. At this point the program is able to draw only orthogonal shapes with arbitrary ratio between their sides.
- The total number of buildings to be presented in the newer time period. This parameter is used to make building distribution more dense or arid depending on the preferences of the user.
- The percentage of changes from the old period to the new one. Supposing that the user requested 200 building in the latest period and a 25% change percentage, the earlier period will depict 150 buildings, thus 50 new buildings have emerged.
- Distribution of buildings in the area of interest. There are two types of distribution supported by the simulation program; grid and random. The grid selection places the

buildings in the nodes of intersecting cells. The number and size of the cells depends on the total area of interest and the number of the buildings. The random choice places the buildings in arbitrary positions, thus they may overlap resulting to more complex shapes.

- Uniform slope of the DTM. The user can define a slope percentage for the DTM on which later the buildings are added to form the DSM.
- Building height. This parameter sets a standard building height for all buildings in the area of interest. The buildings' roof is considered flat. The polygon of each building is layered on the DTM and then it is elevated by this parameter and added to the initial DTM to form the DSM. To keep the roofs flat and not follow the slope of the DTM, the centroid of the building polygon is elevated first and then all the DTM pixels belonging to that building are elevated to that height.
- Radiometric noise. The user is able to set the mean value and standard deviation of a normal distribution from which noise is applied to the images.
- Geometric noise. This parameter can add a uniform geometric noise to the new image in terms of moving a buildings along the x,y axes, rotating it and scaling it with respect to x and y axes.
- DSM noise. Here the user can define the mean value and standard deviation of a normal distribution of noise to be added to the DSM.

To better illustrate the simulating procedure there is a step by step description of the program's execution:

1. An image for the area of interest is created with the dimensions indicated by the user. The color of the image at this point is black.
2. The initial DTM is created based on the given slope in raster format. Each pixel of the DTM corresponds to a single pixel of the initial image.
3. Depending on the selected distribution of buildings, the algorithm generates the centers of each building in the new time period.
4. Given the centers, randomly taking values from the shape list and based on the area of each building, the program calculates the coordinates of the corners of each building with an arbitrary rotation. These values are stored in a container object.
5. The container is copied and depending on the percentage of change some of these polygons are erased to create the set of buildings for the old time period.
6. The polygons of the new buildings are transformed according to the geometric noise selected.
7. The polygons of each period are then drawn of the respective images. The buildings have white outline and white fill.
8. The polygons of each period are superimposed to the DTM and the DSMs are created by elevating building regions of the DTM by the building height selected.
9. Two difference products are created. The first is the image difference and the second the DSM difference. In both cases the old period is differentiated from the new one.
10. Radiometric noise is added to the image difference. This procedure can have one or two steps depending on the needs of the user. In the one step approach the noise generated is simply added to the image. This results in a noisy background, but the white regions remain white because white has the maximum allowed grey tone value and whatever you add to it remains white. To cope with this effect and add noise to the white polygons as

well, it is needed to follow a two step approach. First you add noise as explained before and then you subtract a different noise distribution. This way the background remains noisy and the white polygons get their own amount of noise.

11. The last step is to add noise to the DSM difference. This is achieved by creating a noise distribution according to user provided parameters and add it to the difference DSM.

5. TESTING PROCEDURE

This chapter describes the procedure of testing the proposed strategy with simulated data. For each test run two datasets are created; the first one is used to train the SVM and the second to test its classification performance. To validate the quality of the training the first set is also inputted to the trained SVM. Both sets have the same characteristics in terms of area size and average size of buildings, but the noise parameters may differ as seems fit.

The first step of the procedure is to divide the area in cells of the same size. This size is designated by the average size of each building. It is automatically set to have half the area size of each building. This choice was made in order to secure that a large part of each building will be in a cell. Then for each cell, in each dataset, the algorithm calculates the respective feature vectors from the difference image and the difference DSM. The feature vector may include the mean and standard deviation of gray tone values, from the difference images, and the mean and standard deviation values of heights, from the difference DSMs. The difference data at this point include all the user defined noise.

The next step is to label which cells in each dataset contain changes and mark them as ground truth. To do that automatically, the difference data are used without any radiometric noise. This way all changes have white color. An algorithm runs through the cells and by calculating the mean value of each one decides if the cell contains change. The mean value threshold is set to 140, so for a cell to be considered as changed it must contain about 55% of changed pixels. This threshold ensures that there are enough changed pixels in a cell to calculate reliable feature vectors for changes.

After labeling the SVM is trained and tested. Testing includes both the training set and the test set. For the SVM related operations the library used is libsvm. Libsvm is an open source high end library for SVMs which is considered a standard for such operations and contains precompiled scaling, training and testing tools with a lot of customization options (Chang & Lin, 2001). The final results are exported in terms of changed cells. To have the results also at object level, the changed cells are plotted to the change images and a user overviews them to measure the total success rates. This process can then be repeated with different parameters.

Before presenting the results of some indicative scenarios, there is a brief overview of the testing procedure accompanied with examples from data simulation to change assertion. The first step is to create the images (Figures 4 and 5) and DSMs, if appropriate, of the area of interest. If selected, geometric noise is added to the new image period at this point.

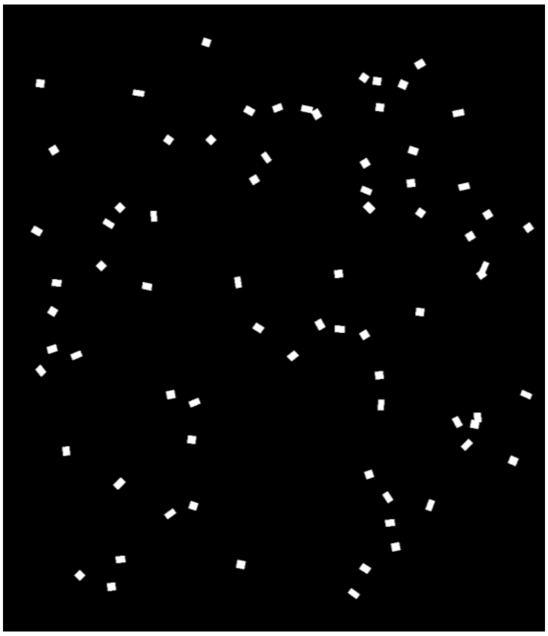

Figure 4. Train set, old time period, with random building distribution

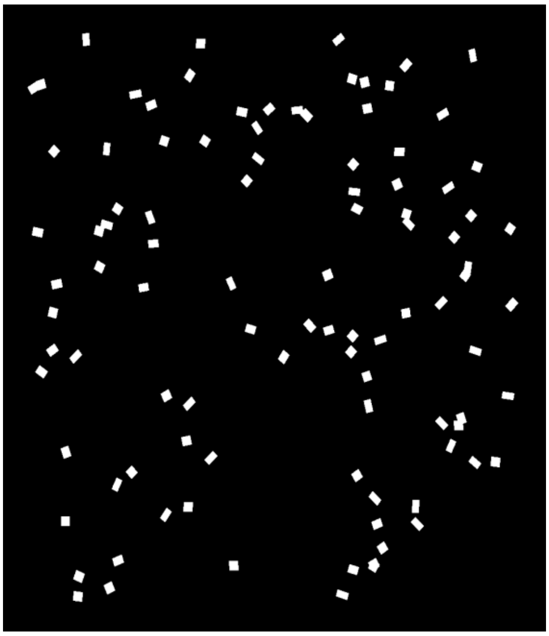

Figure 5. Train set, new time period, with geometric noise

The next step is to produce the difference image (Figure 6) and difference DSM (Figure 8 left). At this point radiometric noise is added to the image difference (Figure 7) and height noise to the difference DSM (Figure 8 right).

Figure 6. Train set's image difference without radiometric noise

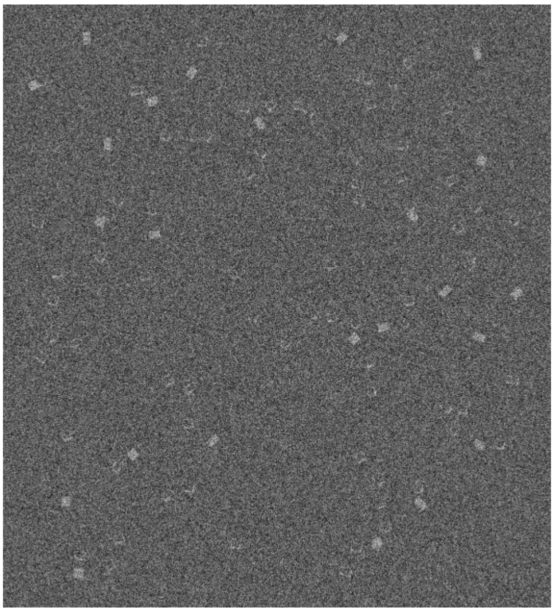

Figure 7. Train set's image difference with radiometric noise

With the base data ready, the feature vectors are extracted for each set and then the SVM is automatically trained. This is followed by the classification of the test and train sets and the assessment of the final results (examples are illustrated in Figures 9 and 10).

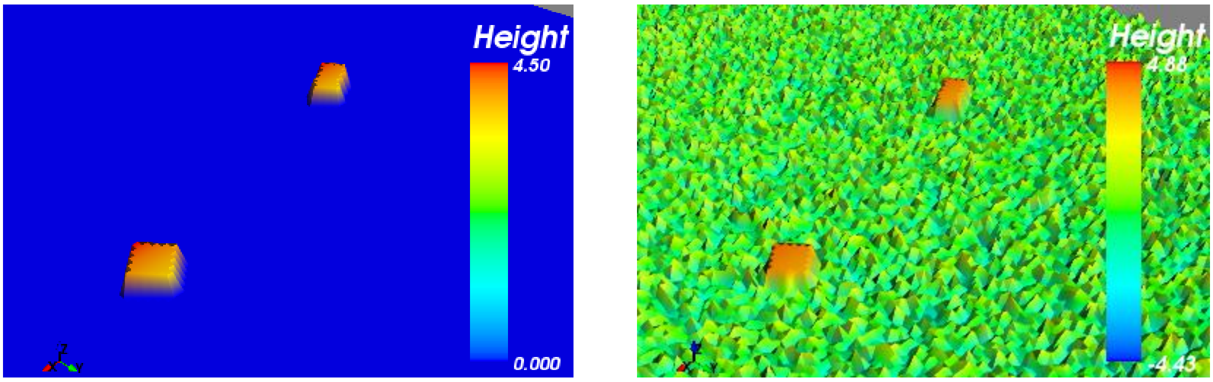

Figure 8. Left: DSM difference without height noise. Right: DSM difference with height noise

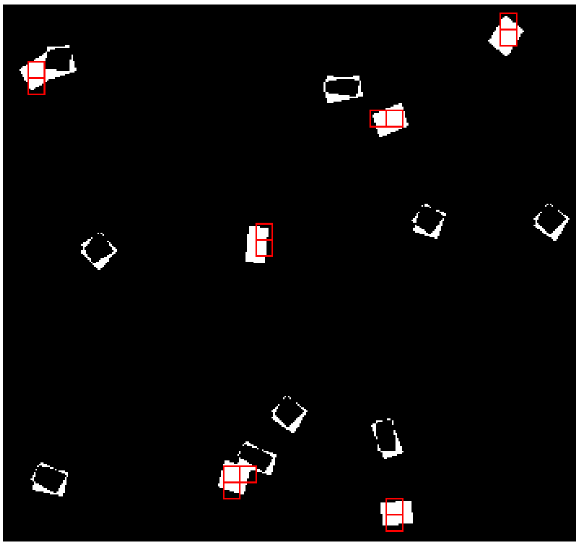

Figure 9. Overlay of the training cells, red, on the change image without radiometric noise

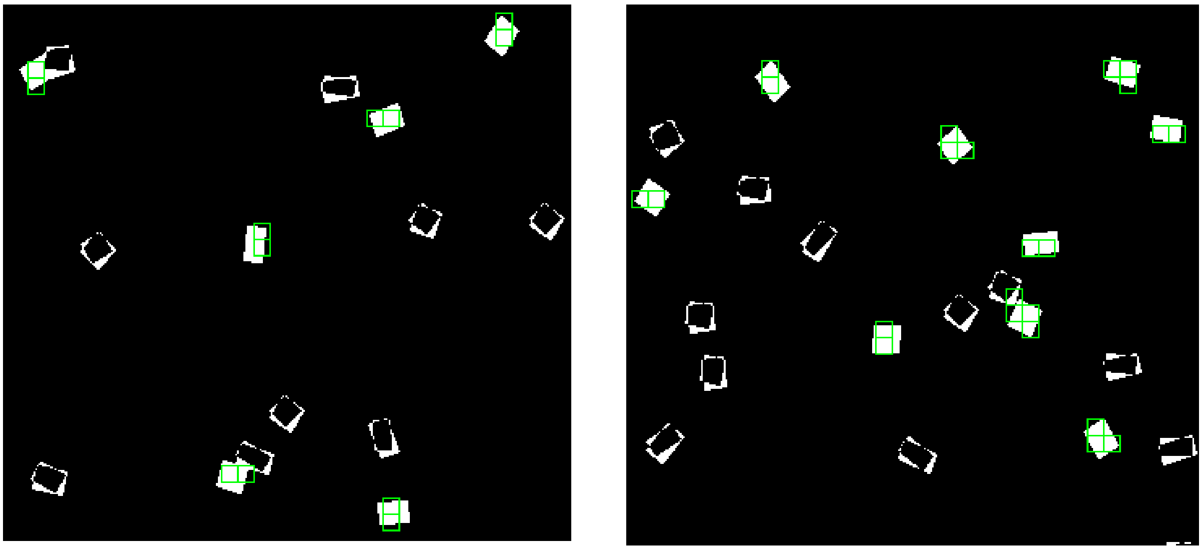

Figure 10. Overlay of returned cells for the training set (left) and the test set (right), green, on the respective change image without radiometric noise

6. RESULTS

This chapter presents some indicative results from testing the above described strategy. For the tests we simulated an area of interest with profound changes. When the buildings' distribution is set to grid the area contains a total of 99 buildings, 74 of which are old ones. When it is set to random, the area contains 100 buildings, 75 of which are old ones. The mean area of the buildings, in all distributions, is 200 pixels. The buildings' shape is orthogonal and the aspect ratio of their sides is determined randomly. The allowed ratios are 1/1, 16/9 and 4/3. Each building is rotated arbitrarily around its center of mass. The imposed rotation is between 0 and 180 degrees. The area of interest is depicted in 1100x1000 pixel images. Whenever a DSM is used, the slope of the original DTM is set to 10% and the average height of each building is set to 4m. The size of the cells to extract features from is 10x10 pixels and all the feature values are scaled to the [-1,1] range. The SVM RBF is used for classification in all the scenarios.

In the following testing scenarios the authors tried out different buildings' distributions, they used images or images and DSMs as data and enforced different types of noise with varying magnitude. In each scenario one train set and one test set are created. Both sets are labeled automatically with the process described in the previous chapter. The SVM is trained on the first one and then classifies the second. Finally, to assess the validity of training, the SVM is also used to classify the train set.

To assess the final results the following are measured:

- accuracy at object level, how many buildings were correctly detected over the actual truth.
- True positive, true negative, false positive and false negative returns at cell level.

There are two types of assessments because the method itself cannot connect in some way cells to objects. Thus a changed building may be represented, and usually is, by more than one changed cells. The role of connecting cells to actual objects is left for the user. User tasks in this case are supported from special imaging products depicting changed cells returned from the algorithm over the noise free difference image.

6.1 Classifying images with radiometric and geometric noise

The first family of test scenarios include different cases of image only data, with geometric and radiometric noise, grid and random distributions of buildings. In every scenario of the Table 1 the geometric noise is fixed to the following values:

- 1 pixel displacement along the x axis and 2 along y axis
- 18 degrees of counter-clock wise rotation
- +10% scale along x axis and +20% along y axis.

The above values are more than those expected in real life situations, but are very helpful to assess the robustness of the method. Radiometric noise varies and it is added to the difference image. The training and testing sets for each case include exactly the same type of noise.

	Scenario 1		Scenario 2		Scenario 3		Scenario 4		Scenario 5		Scenario 6	
	Train	**Test**	**Train**	**Test**	**Train**	**Test**	**Train**	**Test**	**Train**	**Test**	**Train**	**Test**
Mode	Grid		Grid		Grid		Random		Random		Random	
Old buildings	74		74		74		75		75		75	
New buildings	25		25		25		25		25		25	
Mean noise	100		150		200		100		150		200	
Std noise	50		50		30		50		50		30	
Positives	70	76	77	76	72	76	55	55	55	56	55	53
Negatives	10930	10924	10923	10924	10928	10924	10945	10945	10945	10944	10945	10947
True Positives	65	70	71	70	69	70	51	49	53	52	54	51
True Negatives	10930	10924	10923	10924	10928	10924	10945	10945	10944	10944	10943	10945
False Positives	0	0	0	0	0	0	0	0	1	0	2	2
False Negatives	5	6	6	6	3	6	4	6	2	4	1	2
Object accuracy	25/25	25/25	25/25	25/25	25/25	25/25	25/25	24/25	25/25	25/25	25/25	25/25

Table 1. Results of test scenarios, including image only data, with geometric and radiometric noise

From the above it is obvious that, at least at cell level accuracy the random distribution of buildings poses more challenges. It also evident that geometric and radiometric noise, have no significant impact to the results. This is attributed to the fact that white new buildings remain white since radiometric noise is only added to the difference image and thus the buildings remain white. In terms of objects, in almost all cases the new buildings are successfully detected.

6.2 Classifying images and DSMs with two step noise

Following the above, the next phase included testing both with images and DSMs. Only the random mode was tested. Geometric noise was kept at the same levels as above, but this time radiometric noise was generated in two steps, to make the detection procedure more complicated. In the first step a noise distribution is added to the difference image. This makes the black background of the image noisy, but keeps the buildings white and noise free. In the next step a new noise distribution is generated and then subtracted from the first step result. This way both background and buildings get an amount of noise. It is important to generate two noise distributions of different standard deviations in order for the second not to alleviate the effect of the first one. The preferred choice is first to add a distribution with high mean value and low standard deviation and then subtract one with medium mean and high standard deviation. With this strategy the background takes initially the approximate color values of the buildings and then after the subtraction the scene is almost equally randomized. To assess the impact of the DSM to the classification process the same scenarios are ran with and without DSMs. Finally, to further calculate the impact of DSM quality to the results the DSM is first left without noise and then a noise distribution of zero mean value and 1m standard deviation is added to it.

	Scenario 7						Scenario 8					
	Images		Images, DSMs		Images, DSMs		Images		Images, DSMs		Images, DSMs	
	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test
Old buildings	75						75					
New buildings	25						25					
Mean noise 1	180						200					
Std noise 1	10						10					
Mean noise 2	100						100					
Std noise 2	60						60					
Mean height noise	-	-	-	-	0	0	-	-	-	-	0	0
Std height noise	-	-	-	-	1	1	-	-	-	-	1	1
Positives	58	56	58	56	58	56	59	55	59	55	59	55
Negatives	10942	10944	10942	10944	10942	10944	10941	10945	10941	10945	10941	10945
True Positives	50	41	56	52	53	47	50	37	57	50	53	44
True Negatives	10941	10942	10942	10943	10941	10943	10939	10942	10939	10940	10935	10934
False Positives	1	2	0	1	1	1	2	3	1	2	1	2
False Negatives	8	15	2	4	5	9	9	18	2	5	6	11
Object accuracy	23/25	21/25	25/25	24/25	24/25	23/25	22/25	20/25	24/25	24/25	23/25	23/25

Table 2. Results of test scenarios including both images and DSMs, with geometric and radiometric noise

The results of this phase (Table 2) show three important pointers: (a) The two step radiometric noise significantly reduces the accuracy of the process, (b) Considering the DSM in the classification process improves the results (in terms of objects, the successfully detected new buildings are more than 90%), and (c) The DSM quality is proportional to the final accuracy.

6.3 Classifying images and DSMs with low noise training and high noise testing data

To make testing more realistic it was considered that in certain cases training data and the data to be classified differ in quality. To assess this possibility the training data generated were set to have low noise and the data to be classified were set to have higher noise index.

These scenarios were evaluated once taking into account only the images and then adding the DSM as well. The DSM had varying quality whereas the images had a fixed quality.

	Scenario 9				Scenario 10			
	Images		Images, DSMs		Images		Images, DSMs	
	Train	Test	Train	Test	Train	Test	Train	Test
Old buildings	100				100			
New buildings	25				25			
x displacement	1 pixel	1 pixel	1 pixel	1 pixel	1 pixel	1 pixel	1 pixel	1 pixel
y displacement	1 pixel	2 pixel	1 pixel	2 pixel	1 pixel	2 pixel	1 pixel	2 pixel
Rotation	9 deg	18 deg	9 deg	18 deg	9 deg	18 deg	9 deg	18 deg
x scale	+5%	+10%	+5%	+10%	+5%	+10%	+5%	+10%
y scale	+5%	+20%	+5%	+20%	+5%	+20%	+5%	+20%
Loc	180	200	180	200	180	200	180	200
Scale	10	10	10	10	10	10	10	10
Loc2	100	100	100	100	100	100	100	100
Scale2	20	60	20	60	20	60	20	60
DTM scale	-	-	0	0	-	-	0	0
DTM loc	-	-	0.5	1	-	-	0.3	0.3
Positives	45	63	45	63	43	55	43	55
Negatives	10955	10937	10955	10937	10957	10945	10957	10945
True Positives	44	24	44	31	41	15	43	31
True Negatives	10955	10937	10955	10937	10957	10945	10956	10945
False Positives	0	0	0	0	0	0	1	0
False Negatives	1	39	1	32	2	40	0	24
Object accuracy	25/25	18/25	25/25	20/25	25/25	17/25	25/25	21/25

Table 3. Results of test scenarios with low noise training and high noise testing data

From these scenarios it is also evident the positive impact of DSM and high DSM quality to the final result. As it was expected it is very crucial to train and test the SVM in almost the same conditions in order to achieve uniform levels of acceptable accuracy. If this is not possible then a high quality DSM can greatly enhance the overall performance of the algorithm.

7. CONCLUSION

In an effort to improve the procedures for automated and reliable detection of new buildings, which can be used for several applications such as map updating, monitoring of informal development etc, a procedure using Support Vector Machines was developed and tested. The results using the Radical Basis Function SVM classifier are especially encouraging.

For the selection of the appropriate parameters for the application of the RBF SVM method and the investigation of its impacts on the efficiency of the procedure, the use of simulation data is necessary. This is a sophisticated procedure since it is necessary to gradually identify realistic conditions for the application in a virtual environment. However, it is proven that the procedure is especially useful and efficient with interesting results. A significant number of scenarios was tested, the most representative of which are presented in Tables 1-3. The derived results are:

– For cases where simple noise conditions have been applied (even with high radiometric noise), the results were excellent. All new buildings were detected and few false replies (positive or negative) were mentioned. Consequently, the proposed procedure works perfectly under ideal conditions.
– When the inserted noise becomes complicated (two step noise) the success indicators are reduced (e.g., detection of new buildings 80% and 5% false negative responses). The use of DSMs seems to be necessary; DSMs of high quality and accuracy give the best results.
– When training and testing data differ in quality, the difficulties in detection of changes grow. It is proven that the use of images alone is no longer adequate (detection of new buildings is reduced to less than 70%). The use of DSMs is necessary; the increased quality of DSM leads to better results.

Future research should focus on the application of the proposed procedure with real data, aerial and satellite images and DSMs of various densities and quality, so that the results derived from simulation tests will be ensured. However, it is obvious that this method is promising increased percentages of successful detection of new buildings in a broad variety of applications.

REFERENCES

Abe, S., 2005. *Support vector machines for pattern classification*, Springer, pp. 39-40.

Blaschke, T., 2005. Towards a framework for change detection based on image objects. In: *Remote sensing and GIS for environmental studies, Vol. 113*, S. Erasmi, B. Cyffka, and M. Kappas eds., Gottinger Geographische Abhandlungen, pp. 1–9.

Boser, B.E., Guyon, I.M., Vapnik, V., 1992. A Training Algorithm for Optimal Margin Classifiers. In: *Proceedings of the 5th Annual ACM Workshop on Computational Learning Theory*, pp. 144-152.

Burges, C., 1998. A tutorial on support vector machines for pattern recognition. In: *Data mining and knowledge discovery*, vol. 2, pp. 121–167.

Canty, M.J., Nielsen, A.A., 2004. Unsupervised classification of changes in multispectral satellite imagery. In: *Proceedings of SPIE*, pp. 356-363.

Chang, C.C., Lin, C.J., 2001. LIBSVM : A library for support vector machines, available at: http://www.csie.ntu.edu.tw/~cjlin/libsvm (date accessed 3 Sep. 2009)

Cristianini, N., Taylor, S., 2000. *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*, Cambridge University Press.

DTREG, 2009. Support Vector Machines, available at: http://www.dtreg.com/svm.htm. (date accessed 3 Sep. 2009)

Hall, O., 2003. A Multiscale Object-Specific Approach to Digital Change Detection. In: *International Journal of Applied Earth Observation and Geoinformation*, vol. 4, pp. 311-327.

Hofmann, P., Strobl, J., Blaschke, T., Kux, H., 2006. Detecting informal settlements from Quickbird data in Rio de Janeiro using an object based approach. In: *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Salzburg, Austria, Vol. XXXVI, Part 4/C42, unpaginated CD-ROM.

Mo, D., Lin, H., Li, J., Sun, H., Zhang, Z., Xiong, Y., 2008. A SVM-Based Change Detection Method from Bi-Temporal Remote Sensing Images in Forest Area. In: *First International Workshop on Knowledge Discovery and Data Mining*, pp. 209-212.

Moeller, M.S., Blaschke, T., 2006. Urban change extraction from high resolution satellite image. In: *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vienna, Austria, Vol. XXXVI, Part 2, pp. 151-156.

Psaltis, C., Ioannidis, C., 2008. Simple Method for Cost-Effective Informal Building Monitoring, *Surveying and Land Information Science*, vol. 68, pp. 65-79.

Raybaut, P., 2009. Python(x,y), Free scientific and engineering development software, available at: http://www.pythonxy.com (date accessed 3 Sep. 2009).

Song, X., Li, D., 2007. Research on key point of 3D change detection of geospatial database. In: *ISPRS Workshop on updating geo-spatial databases with imagery & 5th ISPRS Workshop on DMGISs*, Urumchi, China, pp. 23-27.

Straub, B.M., Wiedemann, C., Heipke, C., 2000. Towards automatic interpretation of images for GIS update. In: *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Amsterdam, Netherlands, Vol. XXXIII, Part B2, pp. 525-532.

Vapnik, V., Lerner, A., 1963. Pattern recognition using generalized portrait method. In: *Automation and Remote Control*, vol. 24, pp. 774–780.

Wang, H., Pan, L., Zheng, H., Liu, P., 2007. Automatic Detection Technique of Residential Area Change Based on Topological Analysis. In: *ISPRS Workshop on updating geo-spatial databases with imagery & 5th ISPRS Workshop on DMGISs*, Urumchi, China, pp. 29-33.

Zhigao, Y.,Qianqing, Q., Qifeng, Z., 2006. Change Detection in High Spatial Resolution Images Based on Support Vector Machine. In: *IEEE International Symposium on Geoscience and Remote Sensing*, pp. 225-228.

BIOGRAPHICAL NOTES

Christodoulos PSALTIS

Surveying Engineer, PhD student.

Degree of Survey Engineering in 2005 from School of Rural and Surveying Engineering, National Technical University of Athens (NTUA), Greece.

Special research fields: automation in change detection problems, deformation monitoring and close range photogrammetric applications.

Active member of Commission 3 of FIG and Commission 5 of ISPRS.

Charalambos IOANNIDIS

Assistant Professor at the Lab. of Photogrammetry, School of Rural and Surveying Engineering, National Technical University of Athens (NTUA), Greece, teaching photogrammetry and cadastre. Until 1996 he worked at private sector.

1992-1996: Co-chairman of Commission VI -WG2-'Computer Assisted Teaching' in ISPRS.

1997-2001: Member of the Directing Council of Hellenic Mapping and Cadastral Organization and Deputy Project Manager of the Hellenic Cadastre.

His research interests focus on terrestrial photogrammetry, aerial triangulations, digital orthophotos, applications of digital photogrammetry on the cadastre and GIS. He has authored more than 80 papers in the above fields, and has given lectures in related seminars both in Greece and abroad.

CONTACTS

Mr. Christodoulos Psaltis
National Technical University of Athens
Iroon Polytexneiou 9
Zografos, Athens
Greece
Tel. +302107722653
Fax +302107722677
Email: cpsaltis@survey.ntua.gr

Ass. Prof. Charalambos Ioannidis
National Technical University of Athens
Iroon Polytexneiou 9
Zografos, Athens
Greece
Tel. +302107722686
Fax +302107722677
Email: cioannid@survey.ntua.gr