ISSN 1996-3343

Asian Journal of Applied Sciences

http://knowledgiascientific.com

Knowledgia
SCIENTIFIC
A Place to Publish Outstanding Research

Asian Journal of Applied Sciences 4 (1): 63-71, 2011
ISSN 1996-3343 / DOI: 10.3923/ajaps.2011.63.71

Combining Spiking Neural Network with Hausdorff Distance Matching for Object Tracking

Hayat Yedjour, Boudjelal Meftah, Dounia Yedjour and Abdelkader Benyettou
Department of Computer Science, Faculty of Science, University of Science and Technology of Oran Mohammed Boudiaf USTO, S.IM.PA (Signal Image Parole) Laboratory, Mohamed Boudiaf USTO, P.O. Box 1505, El Menaour, Oran, Algeria

Corresponding Author: Hayat Yedjour, Department of Computer Science, Faculty of Science, University of Science and Technology of Oran Mohammed Boudiaf USTO, S.IM.PA (Signal IMage PArole) Laboratory, Mohamed Boudiaf USTO, P.O. BOX 1505, El Menaour, Oran, Algeria

ABSTRACT

This study proposes a new method of extracting and tracking a non_rigid object moving while allowing a static camera. For object extraction we first detect an object using a spiking neural networks for extracting its edge. For object tracking we take this edge as model of the object to localize and match its motion in the next frame by using a Hausdorff distance. The model of object is then updated at each frame along the video sequence. The parameters used are adjusted efficiently along the trajectory of the target to ensure a best track.

Key words: Spiking neural network, integrate-and-fire, edge detection, object tracking, Hausdorff distance

INTRODUCTION

The problem of tracking objects in video has attracted a lot of researchers around the world. Object tracking is so basic and in high demand that it is an indispensable component in many applications including robot vision, video surveillance, object based compression, etc. The problem of object tracking can be divided into two subproblems, extraction of a target object and tracking it over time. In general even those subtasks are not easy because of cluttered backgrounds and frequent occlusions. However, we can readily find a large number of studies on diverse methods using a variety of features such as color, edge, optical flow and contour (Leymarie and Levine, 1993).

Most object recognition systems require a similarity measure between the model or reference features and the image features. The Hausdorff distance measures the divergence of a set of features with respect to a reference set of features. The features are usually image edges in practice. This distance has been used for comparing images as by Huttenlocher *et al.* (1993) and Belogay *et al.* (1997). In general, we are interested in using the Hausdorff distance to identify instances of a model in an image or to track a moving object in a scene.

When object is tracked by edge, this treatment requires an edge map for each frame of the video sequence which reflects the position of the object each time. The edge map must be updated along the video sequence, the computational time needed to compute the edges of each frame must be reduces. Several methods exist for edge detecting in a visual image (Serra, 2006), some are more or less complex and others are more or less greedy in computing times. The application of edge detectors as form as differentiator filters can obtain the edge of objects in the scene, but this kind of techniques does not guarantee closed edges and does not solve the problem of noise and undetected objects.

In this study, the edge map in each image of the sequence are produced by a network model based on spiking neurons (SNN) (Maguire *et al.*, 2006), inspired by the behaviour of biological receptive fields. Simulation results show that the network based on spiking neurons is able to perform edge detection within a time interval of 100 m sec. This processing time is consistent with the human visual system. For object tracking we take the edge map produced by the SNN to determine its motion in the next frame. The current position of the object within the image is determined using a Hausdorff measurement. This approach matches two images and solves the problems of rotation, scaling and translation transformations between these two images by applying the process of minimizing Hausdorff distance twice on the two sets of features (Teng, 2002).

SPIKING NEURAL NETWORK

Generations of artificial neurons: The evolution of connectionist models is well captured in the categorization proposed by Maass, who classifies neural models in three generations (Maas, 1997) depending on the type of neurons that they use. The first generation of artificial neural networks consisted of McCulloch-Pitts threshold neurons, a conceptually very simple model: a neuron sends a binary high signal if the sum of its weighted incoming signals rises above a threshold value. Neurons of the second generation do not use a step- or threshold function to compute their output signals, but a continuous activation function, making them suitable for analog in- and output. Commonly used examples of activation functions are the sigmoid and hyperbolic tangent. The third generation of neural networks uses neurons to pulses that mimic the behavior of real neurons, this generation once again raises the level of biological realism by using individual spikes. This allows incorporating spatial-temporal information in communication and computation, like real neurons do (Ferster and Spruston, 1995). So instead of using rate coding these neurons use pulse coding. Many of these models are already in circulation (Paugam-Moisy, 2006). Spiking Neural networks against classical neural networks, taking into account that the output of real neurons is not characterized by a continuous function, but by a sequence of discrete indivisible elements called spike. This type of neural network know as Spiking Neural Network (SNN) are able to encode time series in spike-trains (Gerstner and Kistler, 2002; Abeles, 2002).

Figure 1 show an example of neurons communicate, the neurons communicate with each other by spikes, which are discrete events. A spiking neuron model describes a transformation from a set of input spike trains into an output spike train (Dayan and Abbott, 2001).

Integrate-and-fire neurons: A spiking model is a model who describes the series of pulses generated by a neuron. A spiking models are often called Integrate-and-fire, The integrate-and-fire model, which is very commonly used in networks of spiking neurons, will be covered after the conceptually more simple and general spike-response model. This model is simple to understand and implement. However, it approximates the very detailed Hodgkin-Huxley model very well, it captures generic properties of neural activity. One of the simplest spiking model is the leaky integrate-and-fire (LIF) model is widely used in computational neuroscience for its relative simplicity (Abbott, 1999), which is described by one state variable, the membrane potential V, governed by the following equation:

$$C\frac{dV}{dt} = -g_L(V - V_0) + I(t) \tag{1}$$

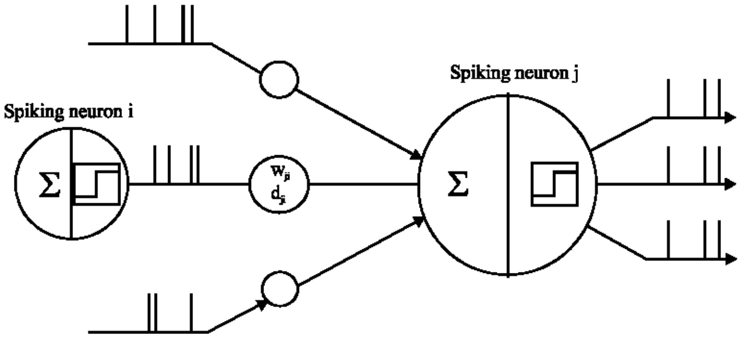

Fig. 1: Example of neurons communicate

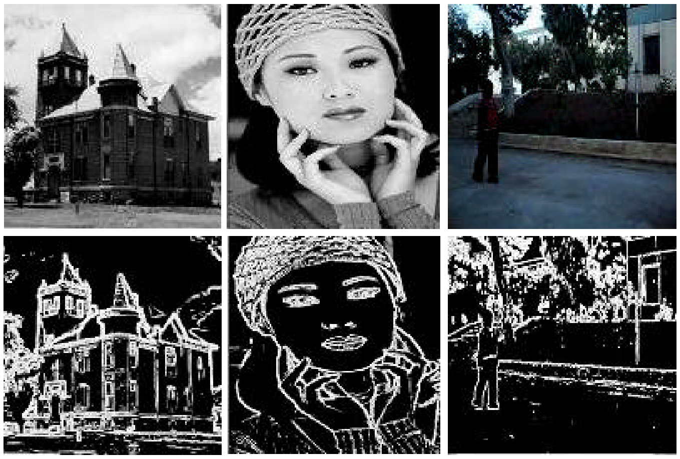

Fig. 2: Examples of Firing rate maps with SNN

where, C is capacitance of the membrane, V_0 is the resting potential (de l'ordre de -70 mV), g_L is leaky conductance and I(t) the input current.

The Eq. 1 models the electrical behavior of the neuron under a threshold, in addition, the firing time t(f) of the neuron is defined by a threshold crossing equation $V(t^{(f)}) = V_{th}$, under the condition $V(t^{(f)})>0$. Immediately after $t^{(f)}$, the potential is reset to a given value V_r. An absolute refractory period d_{abs} can be modelled by forcing the neuron to a value $V \approx V_0$ during a time d_{abs} after a spike emission and then restarting the integration with initial value $V = V_r$.

Spiking neural network model for edge detection: The spiking neural network on witch we have worked (Maguire *et al.*, 2006) is composed of 3 layers: input layer, intermediate layer and output layer. The first layer represents photonic receptors. Each pixel corresponds to a receptor. The intermediate layer is composed of four types of neurons corresponding to four different receptive fields respectively with the synapse connections represents an excitatory synapse and an inhibitory synapse. Each neuron in the output layer integrates four corresponding outputs from intermediate neurons. The firing rate map of the output layer forms an edge graphic corresponding to the input image. The conductance based integrate-and-fire model is applied to the network model. The results of simulation with this network are shown in Fig. 2.

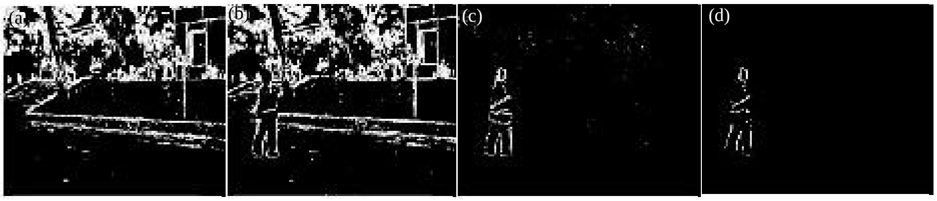

Fig. 3: Extraction of an initial model. (a) Background images SNN, (b) images sequence SNN, (c) Frams differences and (d) Initial model

OBJECT DETECTION

Tracking object requires prior knowledge of the shape and appearance of the objects of interest. For this purpose a model of the object is needed. Several approaches exist for extraction of an initial model for tracking, Kim and Hwang (2002) presented a method for detecting moving objects in a video sequence, their work employs the edge map difference between consecutive frames to extract moving objects. The model is called a segmentation of the moving object from the sequence, it is then tracked in subsequent frames. To simplify and constrain the problem of obtaining an initial model, few assumptions were made: These are that the background is stationary, there is a single moving object, the camera is stationary and there is no occlusion.

Initial segmentation: Initial segmentation is carried out to detect probable moving targets in video sequences. The underlying principle is to take the absolute difference of two consecutive frames. An optimum threshold function is then used to determine the change. If In(u,v) is the intensity of the (u,v)th pixel in the nth frame, then the motion region Mn(u,v) can be extracted by obtaining the difference image of In with In-1 and then thresholding for obtained a binary masks. Pixels of motion are grouped into blobs, regions of connected pixels. The threshold is not yet automated but input by the user.

To avoid false motion indication each individual frame is filtered to remove noise in the difference image by using filters resulting from mathematical morphology which have the property to preserve contours of the image. This method is very sensitive to changes in light or movement of objects in the background but it is faster than the other methods and gives a good result when one moving object must be tracked.

Extraction of an initial model for tracking: The primary aim of object extraction is to locate and delineate the area of an interesting object in a given image. In our case, the target object is extracted using the thresholding difference between two edge maps obtained by SNN, the first representing the background image and the second an image of sequence as shown in Fig. 3a-d.

In this approach, the object model, which is usually in the form of edge map detected by the spiking neural network SNN is continuously updated in each frame of the sequence after the object was detected; i.e., the target model becomes the reference model. This update is recommended to consider only small movements of a non-rigid object between two consecutive frames. The core of this technique is an object tracker that maintains the temporal correspondence of objects throughout the video sequence.

THE HAUSDORFF DISTANCE MEASURES

Hausdorff distance is a scalar measure of the distance between two sets of points. In practice, the two sets of points may be obtained by edge detecting a reference image and a target image to determine the current position of the selected object within the image.

Consider the interpretation of the distance of a single point X from a set of points A. When we say that X is a distance d from A, d is often considered to be the Euclidean distance from X to the nearest point of A. The Hausdorff distance naturally extends this concept to the distance between two sets of points, A and B say. If we determine the distance of each point in B from a set of points A as above, we will then have N Euclidean distance measures, where N is the number of elements in B. Since we want a scalar measure of distance, we choose the maximum value of these distance measures which is known as the directed Hausdorff distance.

The directed hausdorff distance: More formally, the directed Hausdorff distance is defined by:

$$H(A,B) = \max(h(A,B), h(B,A)) \tag{2}$$

$$h(A,B) = \max_{a \in A} \min_{b \in B} \|a - b\| \tag{3}$$

where, A $\{a_1, ..., a_p\}$ and $B = \{b_1, ..., b_q\}$ are two sets of points and $\|.\|$ is the distance between points a and b measured by some norm (generally the Euclidean norm). It identifies the point $a \in A$ that is farthest from any point of B and measures the distance from a to its nearest neighbour in B. Thus h(B, A) is commonly referred to as the forward Hausdorff distance between A and B while h(A, B) is the reverse distance. One interpretation of the forward distance h(A, B) is the distance within which we can find every point of from A and B similarly for the reverse distance. The forward and reverse distances are not necessarily the same.

APPLICATION TO OBJECT TRACKING

Our goal is to track an object in a video sequence by comparing images, once the sequence presented to the SNN network described in earlier, firing rate maps for each sequence image are obtained, each map is then segmented separately to extract moving objects. After performing the segmentation, we obtain binary maps of motion, object models are extracted in each frame as the related areas or blobs each corresponding to a particular movement, then the principle of tracking is as follows: Found in both images I(t-1) and I(t) regions that correspond to the same physical object, person or another, using the Hausdorff distance based approach to localize and match edge models (a reference model with target model) in a video sequence. The overview of the framework is shown in Fig. 4.

Block matching: The block matching can be regarded as a problem of comparing shapes. In our case, tracking is based on models matching: reference model with target model, in which the model associated to the object is searched in the current image.

After object detection, blocks extracted at time t-1 are matched with those extracted at time t, initially we take the largest block at motion sense; i.e., that contains the largest number of pixels, this marked block as X_i becomes the reference_feature (I_{ref}).

Then, for each image sequence, after target detection, we obtain blocks I_k (I_k : target_feature): we must find the block $R_i \in I_k$ with the better similarity to a block $X_i \in I_{ref}$ to ensure a better match, once found, the block R_i becomes the reference (I_{ref} : reference_feature). So each block of current frame is connected to the block of next frame by better similarity.

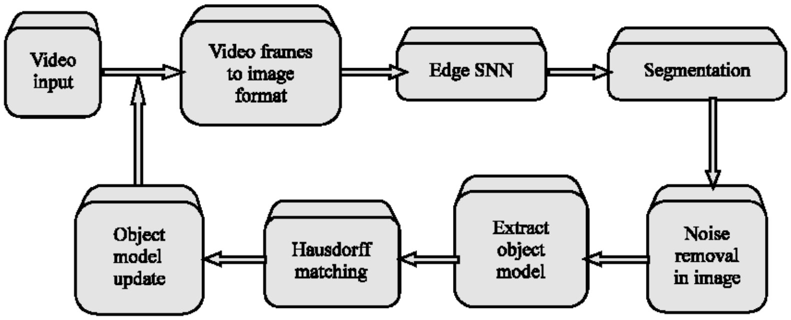

Fig. 4: Process of tracking

Hausdorff distance computations: Once models are extracted, the purpose of this extraction is to measure the similarity between two objects extracted, the first defining the reference model, the second is the target model.

This similarity is defined by a measurement based on the hausdorff distance between two sets of points of edge extracted from the same object in two successive frames of the sequence, if the distance is below a certain threshold τ then we say that the target exists, a rectangle bounding box is drawn using its dimensions and we continue the tracking The position of the object at time t is determined by its position at time t-1 by a better similarity.

There are several techniques for computing the Hausdorff distance and we needed to determine the method best matched to our study for comparison:

that to say X_i I_{ref} , (I_{ref}: image reference; X_i: reference_feature model)
and R_i I_k , (I_k: image cible; R_i: target_feature model)

$$H(X_i, R_i) = \max(h(X_i, R_i), h(R_i, X_i)) \tag{4}$$

$$h(X_i, R_i) = \max_{x \in X_i} \min_{r \in R_i} \|x - r\| \tag{5}$$

We used the directed Hausdorff distance to construct a correlation surface from which the minimum is chosen as the new position of the object, i.e over the distance value is lower, better is the match.

RESULTS AND DISCUSSION

We have proposed the object tracking method using the Hausdorff distance Matching, the tracking is based on edge detection obtained with a spiking neural network. The proposed tracking was implemented on a PC Pentium 4, 3 Ghz system under a Microsoft Windows pack3 simulated by Matlab, It has been tested over a set of video clips recorded with digital camera. The test images have been in BMP format with the dimension of 480×640. The system could track target at approximately 15 frames $\sec^{-1}$. First, we take the background without objects. A red rectangular bounding box is plotted around the object detected from frame difference, The algorithm works very well as long as the object constantly moves. When the object disappears, the box around it goes away. The hausdorff distance between two consecutives frames is calculated and compared against a threshold, here the threshold is decided by the user. If the object stops, this distance becomes zero.

We use this algorithm to test two video sequences, sequence one containing 150 frames and sequence two containing 200 frames, Fig. 5 and 6 show some tracking results of the two video sequences. The proposed algorithm always stably tracks the moving object in these two video sequences.

The moving objects can be reliably tracked until the target goes outside the field of view of the camera . Note that for the simulation we have taken J = 0.85. All the targets are the human movements freely. Figure 5a-j shows a given set of image frames of containing a moving person who walks with a slow movement, Fig. 6a-e shows a given set of image frames of containing a moving person a person which runs a fast movement ,The algorithm detects changes in motion across the two sequence of images and enables robust tracking by using a dynamic template updating.

Influence of the threshold: For test the performance of our algorithm, we will take different values of the threshold J . The same video sequences are applied for the tracking algorithm for different values of the threshold J.

Tracking algorithm using values of HD <J = 0.85. happens to track target at the 150th frame of sequence one and at the 125th frame of sequence two and lost target at last.

For the influence of the threshold, we can conclude that there is one problem with the implementation of the proposed algorithm is that it sometimes fails if an object moves too fast, thus because for big translations, the hausdorff distance was big and therefore exceeds the threshold J.

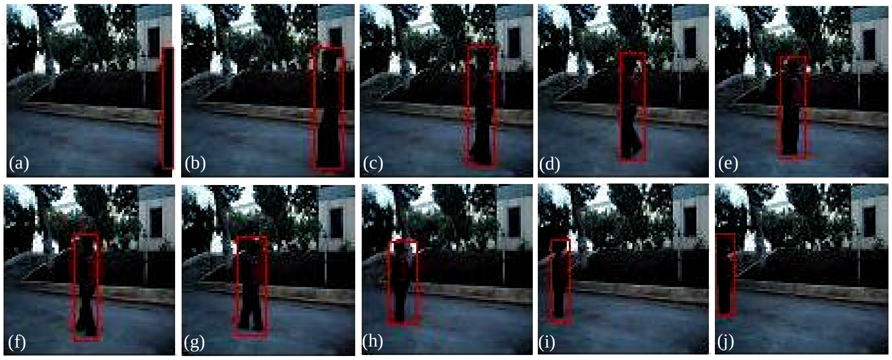

Fig. 5: The experimental result of sequence one. (a) Frame 10, (b) Frame 20, (c) Frame 30, (d) Frame 40, (e) Frame 50, (f) Frame 60, (g) Frame 70, (h) Frame 80, (i) Frame 90, (j) Frame 100

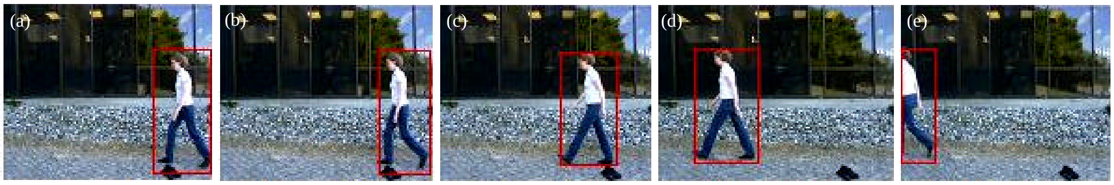

Fig. 6: The experimental result of sequence two. (a) Frame 35, (b) Frame 43, (c) Frame 70, (d) Frame 100 and (e) Frame 160

Table 1: Performance comparison of algorithm in sequence one and two

		HD		SAD	
Sequence	NF	Nrmf	Tr (%)	Nrmf	Tr (%)
One	150	150	100	65	43.33
Two	200	125	62.5	95	47.5

Nrmf : Number of right matching frames, Nf : Number of all frames, Tr : Tracking rate, HD : Hausdorff Distance, SAD : sum of absolute difference

Comparison with other method: To evaluate the performance of this algorithm, the same two video sequences are applied for the tracking algorithms based on Hausdorff distance (HD) and the sum of absolute difference (SAD).

In order to make a comparison between the track algorithm using HD and that using SAD measurements, we take the threshold value at $\tau = 0.85$. Note that the measure, sum of absolute difference is as follows:

$$\text{SAD} = \sum_{i=1}^{n} |X_i - Y_i| \tag{6}$$

The value of this measure decreases as the similarity of intensity values in the windows increase.

Tracking algorithm using the HD happens to track target at the 95th frame of sequence one and at the 86th frame of sequence two and lost target at last. Tracking algorithm using the SAD happens to track target at the 65th frame of sequence one and at the 95th frame of sequence two and lost target at last. The proposed algorithm always stably tracks the moving object in these two image sequences.

The tracking rate is defined as Eq. 7. Table 1 displays the performance of this algorithm for image sequences one and two:

$$\text{Tracking rate (Tr)} = \frac{\text{Number of right matching frames (Nrmf)}}{\text{Number of all frames (Nf) 100\%}} \times 100\% \tag{7}$$

CONCLUSION AND FUTURE WORK

This study proposed an effective application of object tracking with hausdorff distance matching, the tracking is based on edge detection obtained with a spiking neural network. Simulation results show that the network based on spiking neurons is able to perform edge detection within a time interval of 100 msec. This processing time is consistent with the human visual system. Through a series of experimental results, we could confirm that the proposed method was quite stable and produced good results under single non_rigid object and static camera. Particular attention has been given the robustness aspects of our chosen method to ensure a matching of objects and their efficient implementation in terms of computing time using a spiking neural network for edge detection. In the future it could be improved for a robust tracking of multiple targets as the Hausdorff measure has proven its robustness against noise and occlusion, this algorithm can be implemented on other hardware devices and can also be extended for the use of real-time applications and object classifications.

REFERENCES

Abbott, L.F., 1999. Lapique's Introduction of the integrate and fire model neuron (1907). Brain Res. Bull., 50: 303-304.

Abeles, M., 2002. The Handbook of Brain Theory and Neural Networks. 2nd Edn., The MIT Press, Cambridge, MA., ISBN: 978-0262011976. pp: 1-22.

Belogay, E., C. Cabrelli, U. Molter and R. Shonkwiler, 1997. Calculating the housdroff distance between curves. Inform. Process. Lett., 64: 17-22.

Dayan, P. and L.F. Abbott, 2001. Theoretical Neuroscience: Computational and Mathematical Modeling of Neural Systems. The MIT Press, Cambridge, Massachusetts.

Ferster, D. and N. Spruston, 1995. Cracking the neuronal code. Science, 270: 756-757.

Gerstner, W. and W. Kistler, 2002. Spiking Neuron Models: Single Neurons, Pulations, Plasticity. Cambridge University Press, Cambridge.

Huttenlocher, D.P., G.A. Klanderman and W.J. Rucklidge, 1993. Comparing images using the hausdorff distance. IEEE Trans. Pattern Anal. Mach. Intell., 12: 850-863.

Kim, C. and J. Hwang, 2002. Fast and automatic video object segmentation and tracking for contentbased applications. IEEE Trans. Circ Syst. Video Technol., 12: 122-129.

Leymarie, F. and M.D. Levine, 1993. Tracking deformable objects in the plane using an active contour model. IEEE Trans. Pattern Anal. Mach. Intell., 15: 617-634.

Maas, W., 1997. Network of spiking neurons: The third generation of neural network models. Neural Networks, 10: 1659-1671.

Maguire, L.P., B. Glackin, Q.X. Wu, T.M. McGinnity and A. Belatreche, 2006. Learning Mechanism in Networks of Spiking Neurons. Vol. 35. Studies in Computational Intelligence, Springer-Verlag, Berlin/ Heidelberg, ISBN: 978-3-540-36121-3, pp: 171-197.

Paugam-Moisy, H., 2006. Spiking Neuron Networks a Survey. IDIAP Research Institute, University Lyon, France, pp: 2.

Serra, J., 2006. A lattice approach to image segmentation. J. Mathematical Imaging Vision, 24: 83-130.

Teng, Y.C., 2002. Remote-sensing image processing and recognition using wavelet transform and hausdorff distance. Ph.D. Thesis, Institute of Computer Science and Information Engineering National Central University Chung-li, Taiwan.