

NEW INFORMATION TECHNOLOGY OF PERFORMANCE
EVALUATION OF ROAD EXTRACTION FROM
HIGH RESOLUTION SATELLITE IMAGES BASED ON PCNN AND C-V MODEL

NOVA INFORMACIJSKA TEHNOLOGIJA PROCJENE KORISTI
IZDVAJANJA CESTA POMOĆU SATELITSKIH SNIMKI VISOKE REZOLUCIJE
TEMELJENE NA PCNN I C-V MODELU

Ganesh Kumar. T , Murugan D. , Kavitha R. , Manish T.I

*Manonmaniam Sundaranar University, Tirunelveli, India; Unnamalai Institute of Technology, Kovilpatti, India
Sveučilište Manonmaniam Sundaranar, Tirunelveli, Indija; Institut za tehnologiju Unnamalai, Kovilpatti, Indija*

Abstract

Road extraction from high resolution satellite images has been an important research topic for analysis of urban areas. In this paper road extraction based on PCNN and Chan-Vese active contour model are compared. It is difficult and computationally expensive to extract roads from the original image due to presences of other road-like features with straight edges. The image is pre-processed using median filter to reduce the noise. Then road extraction is performed using PCNN and Chan-Vese active contour model. Nonlinear segments are removed using morphological operations. Finally the accuracy for the road extracted images is evaluated based on quality measures.

Sažetak

Izdvajanje cesta pomoću satelitskih slika visoke rezolucije je važna istraživačka tema za analizu urbanih područja. U ovom radu ekstrakcije ceste se uspoređuju na PCNN i Chan-Vese aktivnom modelu. Teško je i računalno skupo izdvojiti ceste iz originalne slike zbog prisutnosti drugih elemenata ravnih rubova sličnih cestama. Slika je pret-hodno obrađena korištenjem filtera za smanjenje smetnji. Zatim se ekstrakcija ceste izvodi pomoću PCNN i Chan-Vese aktivnog modela konture. Nelinearni segmenti su uklonjeni primjenom morfoloških operacija. Konačno, točnost za ceste izdvojene iz slika se ocjenjuje na temelju kvalitativnih mjera.

INTRODUCTION

Automatic detection of roads from aerial and satellite images has been gaining an increasing attention in digital photogrammetry and computer vision. Research differs based on the type and resolution of input images, the primitives employed for road identification, experiment configurations, way of processing, general assumptions, and so on. The major problem to be overcome in this context is the complex structure of the images which contain several different objects such as roads, houses, trees etc. In order to appropriately detect roads from the whole image, the visual characteristics of roads should properly be modeled. An imaged object is defined and

identified by its characteristics, which can be classified in five groups: photometric, geometric, topological, functional and contextual. Examples of these characteristics for an imaged road are:

- Roads are elongated (geometric);
- Road surface usually is homogeneous, at least in a certain portion of the image (photometric);
- Road surface often has a good contrast to its adjacent areas (photometric);
- Roads have a maximum curvature (geometric);
- Roads do not end without a reason (topological);
- Roads intersect and form a network (topological);
- Roads are means of communication between locations (functional);

- Roads may be indicated by a special distribution of trees (contextual);

These characteristics explain many features that are used by a human operator to recognize roads in an image. Especially the functional and contextual characteristics require intelligence in order to make the most of them in the image interpretation process. In practice, the prevailing road properties used in local tests are the spectrum of road materials such as concrete and asphalt, intensity edges at road sides, and the road surface intensity report. These photometric properties are united with the road shape constraints, geometric properties in road finding, tracking and linking procedures.

EXISTING WORK

A new object based road extraction strategy suitable for large scale image maps is elaborated in the paper /1/. The road extraction model is composed of two parallel processes; first one aims to detect straight line segments and the second process is responsible for finding road skeleton. An automatic road detection approach is studied by /2/ based on an edge detection and tracing strategy. A more specific road extraction algorithm proposed in /3/ focuses on the high resolution satellite images, and roads are considered as the regions having continuity and homogeneity properties. Therefore, the authors aim to detect road surfaces rather than road lines. The method suggested in the study /4/ employs mathematical morphology together with active contour model (snakes) to detect roads from high resolution satellite images.

In /5/, the authors combined the input taken from human operator with well-known Bayesian filters such as Kalman and particle filters.

In the research conducted by /6/, an automatic road extraction algorithm which operates on 15 meters LANDSAT Enhanced Thematic Mapper (ETM) panchromatic images is proposed. This study firstly defines the basic characteristics of roads according to their geometric, radiometric, topologic and contextual features. In /7/, another automatic road extraction algorithm using aerial images is proposed. They consider roads as long and thin structures depicting high contrast with respect to their neighborhoods. The features for

this model are extracted by a rule based approach called toe-finding algorithm which mainly analyzes the directional variety of road or non-road footprints. In the Bayesian decision model a lognormal model distribution is constructed by using area-to-perimeter ratios of road footprints. In a more recent study, the authors propose a new method considering two main difficulties faced in the road extraction problem; identifying initial seeds and a robust tracking strategy. In this paper, road network extraction from high resolution satellite images has been developed based on PCNN and C-V Active Contour Model segmentations. The image is first enhanced using median filter. Image smoothing can be achieved in the spatial domain by a low-pass filtering process. Then the resulted image is segmented using PCNN and C-V active contour model. One of the popular approaches is active contour models or snakes, first introduced by Terzopoulos et al. /8/. The initial step is to start with a curve around the object to be detected, the curve moves towards to its interior normal and has to stop on the boundary of the object. The main disadvantage of the original snakes are their sensitivity to initial conditions and the difficulties associated with topological transformations. Caselles et al. /9/ thus introduced the first level set formulation for the geometric active contour model in a non-variational setting and later in a variational form /10/, /11/. A major advantage of the level set approach /12/ is the ability to handle complex topological changes automatically. However, all above active contour models are depend on the gradient of the given image to stop the evolution of the curve. Based on the Mumford-Shah functional /13/ for segmentation, Chan and Vese /14/, /15/ proposed a new level set model for active contours to detect objects whose boundaries are not necessarily defined by a gradient.

Chan-Vese model for active contours /16/ is an efficient and flexible method which is able to segment many types of images, including which would be difficult to segment in means of "classical" segmentation - i.e., using thresholding or gradient based methods. The model is based on an energy minimization problem, which is reformulated in the level set formulation, which makes easier to solve the problem. In 1989, Eckhorn introduced a neural model to emulate the mechanism of cat's visual cortex. The Eckhorn model

provided a simple and effective tool for studying small mammal's visual cortex, and was soon recognized as having significant application in image processing. In 1994, the Eckhorn model was adapted Johnson, who termed this algorithm Pulse-Coupled Neural Network. Over the past decade, PCNNs have been utilized for a variety of image processing applications, including: image segmentation, feature generation, face extraction, motion detection, region growing, noise reduction, and so on. The basic property of the Eckhorn's linking-field model (LFM) is the coupling term. LFM is a modulation of the primary input by a biased offset factor driven by the linking input. These drive a threshold variable that decays from an initial high value. When the threshold drops below zero it is reset to a high value and the process starts over. This is different than the standard integrate-and-fire neural model which accumulates the input until it passes an upper limit and effectively "shorts out" to cause the pulse. This difference causes LFM to be able to sustain pulse bursts, something the standard model does not do on a single neuron level. It is valuable to understand, however, that a detailed analysis of the standard model must include a shunting term due to the floating voltages level in the dendritic compartment(s), and in turn this causes an elegant multiple modulation effect that enables a true higher-order network (HON) /17/ Multidimensional pulse image processing of chemical structure data using PCNN has been discussed by Kinser, et al. /18/.

A simplified PCNN called a spiking cortical model was developed in 2009. /19/ The stimulus for the PCNN is typically a 2D gray scale or color image. The PCNN will transform this input into a series of binary images. The internal feedback among the neurons allows similar pixels, i.e. pixels of similar intensity, to cluster into groups. The PCNN has the ability to map spatial input to temporal output (i.e. convert the input magnitude to a time domain pulses). PCNN are different from classical neural network models, since no training is required. The properties of the PCNN can be adjusted by changing threshold levels and decay time constants. Other interesting features include the ability to detect edges, find texture information, segmentation etc. The same work is done here in this chapter similarly in previous chapter except the segmentation technique here

used is based on PCNN. A segmentation operation followed by morphological operation /20/ renders structural evaluation of the road parts within the multi-resolution analysis framework. The extracted features are fed into performance analysis based on the metrics precision, recall and accuracy. Fig.2.1 depicts the flow-chart of the proposed method. The paper is organized as follows. Section III deals with the algorithm. Experimental results are depicted in Section IV. Performance Analysis is evaluated in Section V. Finally it ends with conclusion in Section VI.

I. THE ALGORITHM

The algorithm has four steps. They are described in details below.

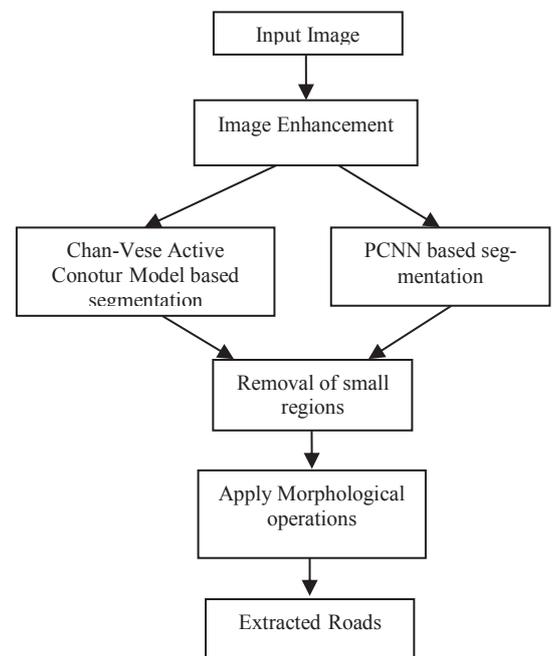


Fig. 2.1 Flowchart of this algorithm

II. DATA PREPROCESSING

ENHANCEMENT IN SPATIAL DOMAIN

Most images are affected to some extent by noise that is unexplained variation in data instability in image intensity whichever uninterpretable or not of interest. Image analysis is often simplified if this noise can be filtered out. In an analogous way filters are used in chemistry to free liquids from suspended impurities by passing them through a layer of sand. Filters provide an aid to visual interpretation of images, and can also

be used as a precursor to further digital image processing.

Most of the methods considered operated on each pixel separately. Filters change a pixel's value taking into account the values of neighbouring pixels. They may either be applied directly to images, or after transformation of pixel values.

Nonlinear filters- that is, all filters which are not linear are more different and difficult to categorize, and are still an active area of research. They are potentially greater than linear filters because they are able to reduce noise levels without simultaneously blurring edges. However, their theoretical foundations are far less secure and they can produce features which are entirely specious. Therefore care must be taken in using them.

NONLINEAR SMOOTHING FILTERS

In filtering to reduce noise levels, linear smoothing filters inevitably blur edges, because both edges and noise are high-frequency components of images. Nonlinear filters are able to simultaneously reduce noise and preserve edges. However:

- There are a incomprehensible number of filters from which to choose,
- They can be computationally expensive to use,
- They can generate spurious features and distort existing features in images.

Therefore they should be used with caution. The simplest, most studied and most widely used nonlinear filter is the moving median. It will be considered in Histogram-based filters, together with similar filters based on the histogram of pixel values in a neighbourhood.

HISTOGRAM-BASED FILTERS

The Moving Median Filter is similar to the moving average filter, except that it produces as output at a pixel the median, rather than the mean, of the pixel values in a square window centered around that pixel. (The median of a set of numbers is the central one, once they have been sorted into ascending order. For example, the median of {1, 0, 4} is 1, whereas the mean is 5/3) For a filter of size $(2m + 1) \times (2m + 1)$ the output is

$$g_{ij} = \text{median} \{f_{i+k,j+l} : k, l = -m, \dots, m\}$$

$$\text{for } i, j = (m + 1), \dots, (n - m).$$

An easily programmed way of computing the moving median filter would be to sort the set of pixel values $\{f_{i+k,j+l} : k, l = -m, \dots, m\}$ into ascending order, then assign to g_{ij} the middle ranked value, and do this independently for every value of i and j from $(m+1)$ to $(n - m)$. This approach is computationally slow and grossly inefficient. Huang, Yang and Tang (1979) presented a fast recursive algorithm for applying the median filter provided that the number of intensity levels of the image is not too great (no greater than 256 for example). It is based on a local histogram which is updated from one neighbourhood of pixel values to the next and from which the median can be extracted without having to sort the set of $(2m + 1)^2$ pixel values.

Their algorithm is as follows.

For each row of the image, $i = (m + 1), \dots, (n - m)$, perform steps 1 to 9:

1. Set $j = m + 1$
2. Calculate the histogram of $\{f_{i+k,j+l} : k, l = -m, \dots, m\}$
3. Set $M =$ median value in the histogram and $N =$ number of pixels in histogram which are less than M
(Note that M and N are simple to compute from the histogram.)
4. Increase j by 1
5. Update the histogram to include $\{f_{i+k,j+m} : k = -m, \dots, m\}$ and exclude $\{f_{i+k,m-1} : k = -m, \dots, m\}$ and simultaneously update N to remain equal to the number of pixels in the histogram which are less than M
6. If $N > \frac{1}{2} (2m + 1)^2$ then reduce M by 1, update N and repeat this step as many times as is necessary until the inequality is no longer satisfied, then go to step 8.
7. If $N + (\text{number of pixels equal to } M) < \frac{1}{2} (2m + 1)^2$ then increase M by 1, update N and repeat this step as many times as is necessary until the inequality is no longer satisfied.
8. Set $g_{ij} = M$

9. Return to step 4 if $j < n-m$.

This algorithm median filter is as fast as linear filters to compute. (Note that it can be applied equally efficiently in a circular window, or any other convex region. The algorithm can also be adapted to find neighbourhood minima and maxima).

Nonlinear filters are not additive: repeated application of a median filter is not equivalent to a single application of a median filter using a different size of window. By making repeated use of a nonlinear smoothing filter with a small window, it is sometimes possible to improve on noise reduction while retaining fine details which would be lost if a larger window was used.

Fig 4.2 b shows the result after 3 X 3 median filter applied to the satellite image.

A. Chan-Vese Active Contour Model Based Segmentation

Let Ω be a bounded open set of R^2 , with $\partial\Omega$ its boundary. Let $u_0 : \Omega \rightarrow R$ be a given image, and $C(s)$ is a piecewise C^1 $[0, 1]$ parameterized a curve.

Let's denote the region inside C as ω , and the region outside C as $\bar{\Omega} \setminus \omega$. c_1 will denote the average pixels' intensity inside C , and c_2 will denote the average intensity outside C (i.e., $c_1 = c_1(C)$, $c_2 = c_2(C)$).

The objective of Chan-Vese algorithm is to minimize the energy functional $F(c_1, c_2, C)$, defined by:

$$F(c_1, c_2, C) = \mu.Length(C) + v.Area(inside(C)) + \lambda_1 \int_{inside(C)} |\mu_0(x, y) - c_1|^2 dx dy + \lambda_2 \int_{outside(C)} |\mu_0(x, y) - c_2|^2 dx dy$$

(2-1)

Where $\mu \geq 0, v \geq 0, \lambda_1, \lambda_2 > 0$ are fixed parameters (should be determined by the user). As suggested by the paper, the preferred settings are $v = 0, \lambda_1 = \lambda_2 = 1$. The term $Length(C)$ could be re-written more generally as $(Length(C))^p$ for $p \geq 1$, but usually $p=1$.

Here, we are looking for c_1, c_2, C that will be the solution to the minimization problem:

$$\inf_{c_1, c_2, C} F(c_1, c_2, C)$$

(2-2)

Level Set formulation

Redefine the problem in level set formalism, instead of searching for the solution in terms of C . In the level set method, $C \subset \Omega$ is represented by the zero level set of some Lipschitz function $\Phi : \Omega \rightarrow R$ s.t.:

$$\begin{cases} C = \partial\omega = \{(x, y) \in \Omega : \Phi(x, y) = 0\} \\ inside(C) = \omega = \{(x, y) \in \Omega : \Phi(x, y) > 0\} \\ outside(C) = \Omega \setminus \omega = \{(x, y) \in \Omega : \Phi(x, y) < 0\} \end{cases}$$

(2-3)

In the following implementation, given a contour C , $\varphi(x, y)$ is defined as the signed distance function from C , where outside C the sign of $\varphi(x, y)$ is negative.

The object is to evolve x, y , when the evolved contour C in each time t is the zero level set of $\Phi(x, y, t)$.

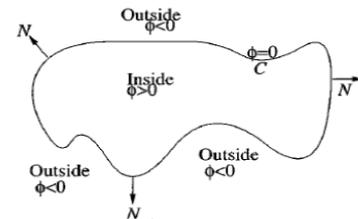


Fig.2.2 The signed distance function

Re-write the functional $F(c_1, c_2, C)$ in terms of $\varphi(x, y)$ only:

$Length(C)$ can be calculated as the length of the zero level set $\varphi(x, y)$:

$$Length(C) = \int_{\Omega} |\nabla H(\Phi(x, y))| dx dy = \int_{\Omega} \delta_0(\Phi(x, y)) |\nabla \Phi(x, y)| dx dy$$

Where $H(z)$ is the Heaviside function:

$$H(z) = \begin{cases} 1, & \text{if } z \geq 0 \\ 0, & \text{if } z < 0 \end{cases}$$

$Area(inside(C))$ can be calculated as the area of the region in which $\varphi(x, y) \geq 0$:

$$Area(inside(C)) = \int_{\Omega} H(\Phi(x, y)) dx dy$$

$\int_{inside(C)} |\mu_0(x, y) - c_1|^2 dx dy$ can be calculated in

terms of $\varphi(x, y)$ when considering only the region in which $\varphi(x, y) > 0$

$$\int_{inside(C)} |\mu_0(x, y) - c_1|^2 dx dy = \int_{(x, y) : \Phi(x, y) > 0} |\mu_0(x, y) - c_1|^2 H(\Phi(x, y)) dx dy$$

In a similar way:

$$\int_{\text{outside}(C)} |u_0(x, y) - c_2|^2 dx dy = \int_{(x, y): \Phi(x, y) > 0} |u_0(x, y) - c_2|^2 dx dy = \int_{\Omega} |u_0(x, y) - c_2|^2 H(\Phi(x, y)) dx dy$$

The average intensities:

$$c_1 = \frac{\int_{\Omega} u_0(x, y) H(\Phi(x, y)) dx dy}{\int_{\Omega} H(\Phi(x, y)) dx dy}$$

$$c_2 = \frac{\int_{\Omega} u_0(x, y) H(1 - \Phi(x, y)) dx dy}{\int_{\Omega} H(1 - \Phi(x, y)) dx dy}$$

The above equations leads to the energy functional in terms of (c_1, c_2, φ) where $c_1 = c_1(\varphi)$, $c_2 = c_2(\varphi)$ and $\delta_0(x)$ is a Dirac delta function:

$$F(c_1, c_2, \Phi) = \mu \int_{\Omega} \delta_0(\Phi(x, y)) |\nabla \Phi(x, y)| dx dy + \nu \int_{\Omega} H(\Phi(x, y)) dx dy + \lambda_1 \int_{\Omega} |u_0(x, y) - c_1|^2 H(\Phi(x, y)) dx dy + \lambda_2 \int_{\Omega} |u_0(x, y) - c_2|^2 H(1 - \Phi(x, y)) dx dy$$

(2-4)

Observe the terms in equation 3-4, the evolution of the curve is influenced by two terms (ν is usually set to 0, so we will ignore it): the curvature regularizes the curve and makes it smooth during evolution; the "region term" $-\lambda_1(u_0 - c_1)^2 + \lambda_2(u_0 - c_2)^2$ affects the motion of the curve [5].

The term $\mu \int_{\Omega} \delta_0(\Phi(x, y)) |\nabla \Phi(x, y)| dx dy$ is

the penalty on the total length of the curve C. For example, if the boundaries of the image are quite smooth, give μ a larger value, to prevent C from being a complex curve.

λ_1, λ_2 affect the desired uniformity inside C and outside C, respectively. For example, Set $\lambda_1 < \lambda_2$ when we expect an image with quite uniform background and varying grayscale objects in the foreground.

Using Euler-Lagrange equations and the gradient-descent method, it is shown in the paper that $\varphi(x, y)$ which minimizes the energy F (c_1, c_2, φ) satisfies the PDE (t is an artificial time):

$$\begin{cases} \frac{\partial \Phi}{\partial t} = \delta(\Phi) \left(\mu \cdot p \cdot \left(\int_{\Omega} \delta(\Phi) |\nabla \Phi| dx dy \right)^{p-1} \operatorname{div} \left(\frac{\nabla \Phi}{|\nabla \Phi|} \right) - \nu - \lambda_1 (u_0 - c_1)^2 + \lambda_2 (u_0 - c_2)^2 \right) \\ = \delta(\Phi) \left(\mu \cdot p \cdot \left(\int_{\Omega} \delta(\Phi) |\nabla \Phi| dx dy \right)^{p-1} \cdot \kappa(\Phi) - \nu - \lambda_1 (u_0 - c_1)^2 + \lambda_2 (u_0 - c_2)^2 \right) \\ \Phi(x, y, 0) = \Phi_0(x, y), \frac{\delta(\Phi)}{\nabla \Phi} \frac{\partial \Phi}{\partial n} = 0 \text{ on } \partial \Omega \end{cases}$$

(2-5)

Where $\kappa(\Phi)$ is the curvature of the evolving curve (for some specific height level in Φ). The curvature can be calculated using the spatial derivatives of Φ upto second order:

$$\kappa(\Phi) = \frac{\Phi_{xx} \Phi_y^2 - 2 \Phi_{xy} \Phi_x \Phi_y + \Phi_{yy} \Phi_x^2}{(\Phi_x^2 + \Phi_y^2)^{3/2}}$$

(2-6)

Numerical scheme

First, define regularizations of H(x) and $\delta(x)$ (where $\delta(x) = \frac{d}{dx} H(x)$):

$$H_{\varepsilon}(x) = \frac{1}{2} \left(1 + \frac{2}{\pi} \arctan \left(\frac{x}{\varepsilon} \right) \right)$$

$$\delta_{\varepsilon}(x) = \frac{1}{\pi} \frac{\varepsilon}{\varepsilon^2 + x^2}$$

(2-7)

For some constant $\varepsilon > 0$. The values used in the simulations are $\varepsilon = h = 1$, where h is the space step (it is reasonable to choose $\varepsilon = h$, since h is the smallest space step in the problem). These regularizations achieve good results in simulations, as described in the paper, that they normally lead to the global minimum of the energy.

Let's define $\Phi_{i,j}^n = \Phi(n\Delta t, x_i, y_j)$ where Δt is the time step. The PDE can be discretized by using the following notations for spatial finite differences (where $h_x = h_y = h$):

$$\Delta_x^- \Phi_{i,j}^n = (\Phi_{i,j}^n - \Phi_{i-1,j}^n) / h,$$

$$\Delta_y^- \Phi_{i,j}^n = (\Phi_{i,j}^n - \Phi_{i,j-1}^n) / h,$$

$$\Delta_x^+ \Phi_{i,j}^n = (\Phi_{i+1,j}^n - \Phi_{i,j}^n) / h$$

$$\Delta_y^+ \Phi_{i,j}^n = (\Phi_{i,j+1}^n - \Phi_{i,j}^n) / h$$

The linearized, discretized PDE becomes:

$$\frac{\Phi_{i,j}^{n+1} - \Phi_{i,j}^n}{\Delta t} = \delta_h(\Phi_{i,j}^n) \frac{\mu}{h^2} \left[\Delta_x^+ \left(\frac{\Delta_x^+ \Phi_{i,j}^n}{\sqrt{(\Delta_x^+ \Phi_{i,j}^n)^2 / h^2 + (\Phi_{i,j+1}^n - \Phi_{i,j-1}^n)^2 / (2h)^2}} \right) + \Delta_x^- \left(\frac{\Delta_x^- \Phi_{i,j}^n}{\sqrt{(\Delta_x^- \Phi_{i,j}^n)^2 / h^2 + (\Phi_{i,j+1}^n - \Phi_{i,j-1}^n)^2 / (2h)^2}} \right) \right] - \delta_h(\Phi_{i,j}^n) \left[\nu + \lambda_1 (u_{0,j} - c_1(\Phi^n))^2 - \lambda_2 (u_{0,j} - c_2(\Phi^n))^2 \right] \tag{2-8}$$

Defining the following constants (for a given Φ^n):

$$C_1 = \frac{1}{\sqrt{(\Phi_{i+1,j}^n - \Phi_{i,j}^n)^2 + (\Phi_{i,j+1}^n - \Phi_{i,j-1}^n)^2 / 4}}, C_2 = \frac{1}{\sqrt{(\Phi_{i,j}^n - \Phi_{i-1,j}^n)^2 + (\Phi_{i,j+1}^n - \Phi_{i,j-1}^n)^2 / 4}}$$

$$C_3 = \frac{1}{\sqrt{(\Phi_{i+1,j}^n - \Phi_{i-1,j}^n)^2 / 4 + (\Phi_{i,j+1}^n - \Phi_{i,j-1}^n)^2}}, C_4 = \frac{1}{\sqrt{(\Phi_{i+1,j-1}^n - \Phi_{i-1,j-1}^n)^2 / 4 + (\Phi_{i,j}^n - \Phi_{i,j-1}^n)^2}}$$

We get the simplified equation:

$$\Phi_{i,j}^{n+1} \left[1 + \frac{\Delta t}{h} \delta_h(\Phi_{i,j}^n) \mu (pL(\Phi^n)^{p-1}) (C_1 + C_2 + C_3 + C_4) \right] = \Phi_{i,j}^n + \frac{\Delta t}{h} \delta_h(\Phi_{i,j}^n) \mu (pL(\Phi^n)^{p-1}) (C_1 \Phi_{i+1,j}^n + C_2 \Phi_{i-1,j}^n + C_3 \Phi_{i,j+1}^n + C_4 \Phi_{i,j-1}^n) - \Delta t \delta_h(\Phi_{i,j}^n) \left[\nu + \lambda_1 (u_{0,j} - c_1(\Phi^n))^2 - \lambda_2 (u_{0,j} - c_2(\Phi^n))^2 \right]$$

(2-9)

Using the regularized Heaviside function calculates $c_1(\Phi^n), c_2(\Phi^n)$ as discretized sums. In addition, $L(\Phi^n)$ is Length(C) as was calculated. In order to solve equation 3-10, we can use an iterative way, as pointed out by [21], proposition 6.1, in which it is also proved that there exists a solution to the equation.

Reinitialization of Φ

In each step, need to reinitialize $\Phi(x, y)$ to be the signed distance function to its zero level set. This procedure prevents the level set function from becoming too "flat", an effect is caused by using the regularized delta function $\delta_\epsilon(x)$, which causes blurring.

The reinitialization process is made by using the following PDE:

$$\begin{cases} \frac{\partial \Psi}{\partial \tau} = \text{sign}(\Phi(x, y, t)) (1 - |\nabla \Psi|) \\ \Psi(0) = \Phi(x, y, t) \end{cases} \tag{2-10}$$

The solution of this equation, Ψ , will have the same zero level set as $\Phi(x, y, t)$, and away from this level set, $|\nabla \Psi|$ will converge to 1, as it should be for a distance function.

The numerical equation for equation 11:

$$\Psi_{i,j}^{n+1} = \Psi_{i,j}^n - \Delta t \text{sign}(\Phi(x, y, t)) G(\Psi_{i,j}^n) \tag{2-11}$$

Where the "flux" $G(\Psi_{i,j}^n)$ is defined using the notations a, b, c, d, defined by:

$$a = (\Delta_x^- \Psi_{i,j}) / h = (\Psi_{i,j} - \Psi_{i-1,j}) / h$$

$$b = (\Delta_x^+ \Psi_{i,j}) / h = (\Psi_{i+1,j} - \Psi_{i,j}) / h$$

$$c = (\Delta_y^- \Psi_{i,j}) / h = (\Psi_{i,j} - \Psi_{i,j-1}) / h$$

$$d = (\Delta_y^+ \Psi_{i,j}) / h = (\Psi_{i,j+1} - \Psi_{i,j}) / h$$

and

$$G(\Psi_{i,j}) = \begin{cases} \sqrt{\max((a^+)^2, (b^-)^2) + \max((c^+)^2, (d^-)^2)} - 1, \Phi(x_i, y_j, t) > 0 \\ \sqrt{\max((a^-)^2, (b^+)^2) + \max((c^-)^2, (d^+)^2)} - 1, \Phi(x_i, y_j, t) < 0 \\ 0, \text{otherwise} \end{cases}$$

(2-12)

where $a^+ = \max(a, 0), a^- = \min(a, 0)$ and so on.

Summary of the algorithm

1. Initialize $\Phi_{i,j}^{n=0}$ to some Lipschitz function Φ_0
2. Compute $c_1(\Phi^n), c_2(\Phi^n)$
3. Solve the PDE of equation 9
4. Reinitialize $\Phi_{i,j}^{n+1}$ as be the signed distance function to $\{\Phi_{i,j}^{n+1} = 0\}$ by using equation 11
5. Check if the solution is stationary. If not, continue. Else, stop.

The process should be stopped when $Q = \sum_{|\Phi_{i,j}^n| < h} |\Phi_{i,j}^{n+1} - \Phi_{i,j}^n| \leq \Delta t \cdot h^2$ (this should be checked at stage no. 5) where M is the number of grid points which satisfy $|\Phi_{i,j} < h|$, because $\Phi(x, y, t)$ is not expected to change anymore (except for maybe some small numerical changes).

The result of this algorithm is shown in Fig.4.2c. Initializing a small curve inside the re-

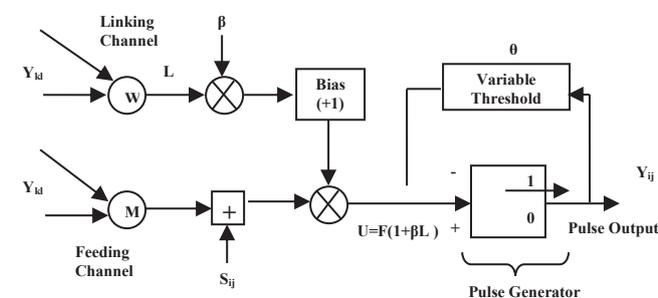
gion of interest shown in Fig 4.2d and allowing it to grow outwards until it reaches the desired boundary as in Fig.4.2e- Fig.4.2i Finally, a segmented image with closed boundary was obtained as shown in Fig.4.2j.

B. Pulse Coupled Neural Network Based Segmentation

A PCNN is a neural network algorithm that, when applied to image processing, gives a series of binary pulsed signals, each associated to one pixel or to a cluster of pixels. It belongs to the class of unsupervised artificial neural networks in the sense that it does not need to be trained. The network consists of nodes with spiking behaviour interacting with each other within a pre-defined grid. The architecture of the network is simpler than other neural network implementations: there are no multiple layers that pass information to each other. PCNNs only have one layer of neurons, which receives input directly from the original image, and form the resulting pulse image. The feeding compartment receives both an external and a local stimulus, whereas the linking compartment only receives the local stimulus. The third compartment is represented by an active threshold value. When the internal activity becomes larger than the threshold the neuron fires and the threshold sharply increases. Afterwards, it begins to decay until once again the internal activity becomes larger. Such a process gives rise to the pulsing nature of the PCNN. Fig.4.4 illustrates the block architecture of PCNN.

More formally, the system can be defined by the following expressions:

Fig.2.6. Schematic representation of a PCNN neuron



ron

$$F_{ij}[n] = e^{\alpha F} \cdot F_{ij}[n-1] + S_{ij} + V_F \sum_{kl} M_{ijkl} Y_{kl}[n-1] \quad (2-13)$$

$$L_{ij}[n] = e^{\alpha L} \cdot L_{ij}[n-1] + V_L \sum_{kl} W_{ijkl} Y_{kl}[n-1] \quad (2-14)$$

where αF and αL is a decay constant, S_{ij} is the input to the neuron (ij) belonging to a 2D grid of neurons, F_{ij} is the value of the feeding compartment and L_{ij} is the corresponding value of the linking compartment. Each of these neurons communicates with neighbouring neurons (kl) through the weights given by M and W respectively. M and W traditionally follow very symmetric patterns and most of the weights are zero. Y indicates the output of a neuron from a previous iteration $[n - 1]$. All compartments have a memory of the previous state, which decays in time by the exponent term. The constant V_F and V_L are normalizing constants. The state of the feeding and linking compartments are combined to create the internal state of the neuron, U . The combination is controlled by the linking strength, β .

The internal activity is given by:

$$U_{ij}[n] = F_{ij}[n](1 + \beta L_{ij}[n]) \quad (2-15)$$

The internal state of the neuron is compared to a dynamic threshold, Θ , to produce the output, Y , by:

$$Y_{ij}[n] = \begin{cases} 1, & \text{if } U_{ij}[n] > \theta_{ij}[n], \\ 0 & \text{otherwise} \end{cases} \quad (2-16)$$

The threshold compartment is described as:

$$\theta_{ij}[n] = e^{-\alpha \theta} \cdot \theta_{ij}[n-1] + V_{\theta} Y_{ij}[n] \quad (2-17)$$

Where V_{θ} is a large constant generally more than one order of magnitude greater than the average value of U .

The algorithm consists of iteratively computing Equation 4-1 through Equation 4-5 until the user decides to stop. Each neuron that has any stimulus will fire at the initial iteration, creating a large threshold value. Then, only after several iterations the threshold will be small enough to allow the neuron to fire again. The result of this algorithm is shown in Fig.4.3a

C. Removal Of Small Regions

After segmentation, the image can be viewed as road and non road features where the road is an elongated region contain maximum area and other non road features such as trees buildings, houses have minimum area. In this module, image which containing minimum region area is set to false and maximum region area is set to true to retain the road features. The result of this module is shown in Fig. 4.2k , 4.3b.

Morphological Operations

By making any pixel less than T as 0 to detect roads, there are chances of detecting dark vehicles, shadows, and trees, which have intensity values less than T (i.e. range similar to that of roads). It can be observed that roads are dark in color. It can also be observed that there are some isolated black pixels, which correspond to dark vehicles, shadows, and trees. To minimize the detection of shadows, dark vehicles, and trees, apply morphological operations: Clean, Dilate, Close, and Thin to the binary image.

Clean: Remove isolated pixels (1's surrounded by 0's).

Dilate: Pixels beyond the image border are assigned the minimum value afforded by the data type. For binary images, these pixels are assumed to be set to 0. For grayscale images, the minimum value for uint8 images is 0.

Close: Dilates an image and then erodes the dilated image using the same structuring element for both operations.

Thin: It removes pixels so that an object without holes shrinks to a minimally connected stroke, and an object with holes shrinks to a connected ring halfway between each hole and the outer boundary.

These morphological operations are inbuilt functions of MATLAB Image Processing Toolkit.

As a result of these four morphological operations, most of the isolated dark pixels will be eliminated. In the particular case where large areas are covered by trees and vegetation (with intensity values in the range of roads), the possibilities of being eliminated are minimal. This is due to the fact that these large areas are not isolated set of pixels and the morphological operations, which are restricted to a 3-by-3 block of pixels, will not eliminate them. If try to increase the size

of the block, there are possibilities of missing the roads. Fig.4.2-l,m and Fig.4.3-c,d show the binary image obtained after applying the morphological operations.

IV PERFORMANCE ANALYSIS

A. Introduction

Human beings play an essential role in evaluating the quality of image segmentation. The subjective evaluation has been long regarded as the most reliable assessment of the segmentation quality. In addition, even for segmentations which are visually close, different human observers may give inconsistent evaluations. As an alternative, the objective evaluation methods, which aim to predict the segmentation quality accurately and automatically, are much more expected.

The existing objective evaluation methods can be classified as ground-truth based ones and non-ground-truth based ones. In non-ground-truth based methods, the empirical goodness measures are proposed to meet the heuristic criteria in the desirable segmentations. Then the score is calculated based on these criteria to predict the quality of segmentation.

The ground-truth based methods measure the difference between the segmentation result and the human-labeled ground truths. They are more intuitive than the empirical based measures, since the ground truths can well represent the human-level interpretation of an image.

Considering only the region boundaries, these measures are more sensitive to the dissimilarity between the segmentation and the ground truths than the region based measures. Some other measures use non-parametric tests to count the pairs of pixels that belong to the same region in different segmentations.

In this work, focus on evaluating segmentation results with ground truth. The existing methods of this kind prefer matching the given whole segmentation with ground truths for evaluation. However, the available human-labeled ground truths are only a small fraction of all the possible interpretations of an image. The available dataset of ground truths might not contain the desired ground truth which is suitable to match the input segmentation. Hence such kind of comparison often leads to a certain bias on the result or is far from the goal of objective evaluation.

To design a segmentation quality measure this could generalize the configurations of the segmentation and preserve the structural consistency across the ground truths. To measure the quality of the segmentation, a composite ground truth can be adaptively produced to locally match the segmentation as much as possible. It will integrate all these factors for a robust evaluation. Fig.3.1 illustrates the flowchart of the proposed framework [22].

Firstly a new composite ground truth is adaptively constructed from the ground truths in the database, and then the quantitative evaluation score is produced by comparing the input segmentation and the new ground truth.

The standard similarity measures, such as Precision, Recall and Accuracy compare the segmentation with the whole ground truths, producing a matching result based on the best or the average score.

Ground-Truth

The object extraction was performed by marking pixels on the object border using a graphics tablet, and subsequently filling the object interior. The result is a series of binary masks, one for each object in the dataset, where zero valued pixels denote the background and non-zero valued pixels denote the object. Creating a 100% pixel accurate ground-truth is, in general, impossible, due to the ambiguity in the true positions of the border pixels. It is necessary, however, when creating a binary ground-truth to decide which pixels belong to the object and which pixels belong to the background. To handle this ambiguity in the object border pixels, a simple heuristic was applied: retain pixels that appear to contain some of the objects color along the object border, and that do not appear to be image compression artifacts. This heuristic was chosen so that each pixel along the border would be, on average half-inside and half-outside the true form of the foreground object [23]. Fig.3.2 shows the ground truth of the test image.

B. Performance Metrics

1. Segmentation Accuracy

The percentage of segmentation accuracy can be defined as [55],

$$\%SegmentationAccuracy = \frac{\text{Number of correctly classified pixels for segmented area}}{\text{Total number of pixels}} \quad (5-1)$$

2. Precision-Recall Measures

The precision and recall values used to characterize the agreement between the oriented boundary edge elements (termed edgels) of region boundaries of two segmentations. Given two segmentations, S and R, where S is the result of segmentation and R is the ground truth, precision is proportional to the fraction of edgels from S that matches with the ground truth R, and recall is proportional to the fraction of edgels from R for which a suitable match was found in S.

Precision measure is defined as follow

$$P = \frac{\text{Matched}(S,R)}{|S|} \quad \text{Recall} = \frac{\text{Matched}(R,S)}{|R|} \quad (5-2)$$

Where |S| and |R| are the total amount of boundary pixels in probabilistic terms, Precision is the probability that the result is valid, and recall is the probability that the ground truth data was detected. A low recall value is typically the result of under-segmentation and indicates failure to capture salient image structure. Precision is low when there is significant over-segmentation, or when a large number of boundary pixels have greater localization errors than some threshold δ_{max} . However, since these measures are not tolerant to refinement, it is possible for two segmentations that are perfect mutual refinements of each other to have very low precision and recall scores [24].

V EXPERIMENTAL RESULTS

The experiment is conducted in the image using the algorithms PCNN and Chan-Vese active contour model and their results shown in Fig.4.1.with required statistical parameters and their results are presented in Table 4.1. If the value of accuracy is higher then the segmentation approach is better.

A. Performance Evaluation

The Performance analysis chart shown in Fig.4.1.reveals that the accuracy of PCNN is higher than Chan-Vese active contour method and

also the precision and recall are higher than it. Comparatively the PCNN algorithm provides good result. So it was observed that the method PCNN performs better compare to Chan-Vese active contour approach.

VI CONCLUSION

By using the two algorithms, better results are obtained for Pulse coupled neural network based segmentation. An active contours based on techniques of curve evolution, Chan-Vese algorithm for segmentation and level sets is a good and accurate method to detect object (region) boundaries, to isolate and extract individual components from our digital image. It is possible to detect objects whose boundaries are not necessarily defined by gradient by, where the stopping term does not depend on the gradient of the image, as in the classical active contour.

The Chan-Vese algorithm for image segmentation and its result shown that it is effective on a segmentation of image. It is especially useful in cases where an edge-based segmentation algorithm will not suffice, since it relies on global properties (graylevel intensities, contour lengths, region areas) rather than local properties such as gradients. This means that it can deal gracefully with noisy images, blurry images, and images where the foreground region has a complicated topology (multiple holes, disconnected regions, etc).

Although the test image took under a minute to segment, the Chan-Vese algorithm is prohibitively slow for some applications. Depending on the type and size of the image and the number of iterations needed, the segmentation can take several seconds, which is too slow to keep up with typical video frame rates. A segmentation method based on a model of the pulse coupled neural network was implemented and tested on satellite image. Certain adaptations were necessary in order to use the PCNN as an image analysis tool. These adaptations include the modification of the neuron's feeding field, the single pulse neuron, single time step linking, and a single threshold decay variable used by all neurons. These modifications to the model neuron resulted in over an order of magnitude performance increase as well as reducing the memory requirements by over thirty per-

cent. This work clearly demonstrates that the PCNN can be used as an effective image segmentation tool compared to Chan-Vese active contour model.

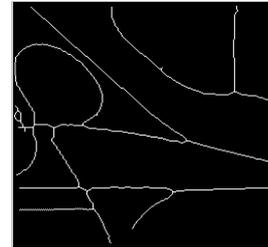


Fig.3.2 Ground Truth Image

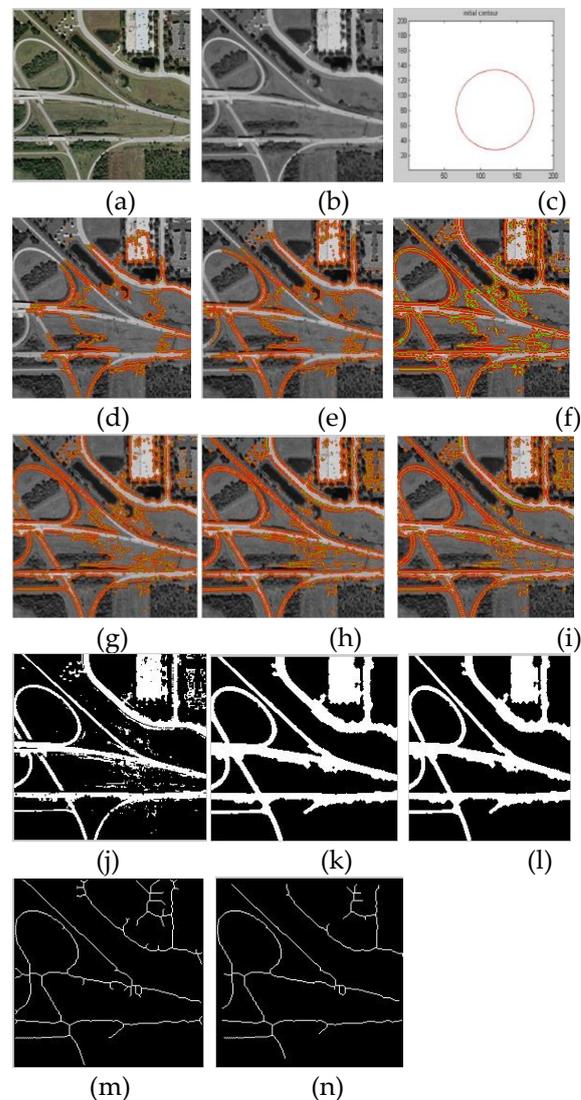


Fig. 4.2 a. Original Image b. Enhanced Image c. Initial Contour d, e, f, g, h, i. After 160, 260, 460, 560, 1280, 1500 iterations respectively j. Segmentation result of C-V model k. Result after removal of

small regions l, m. Result after applying morphological operations n. Extracted Roads

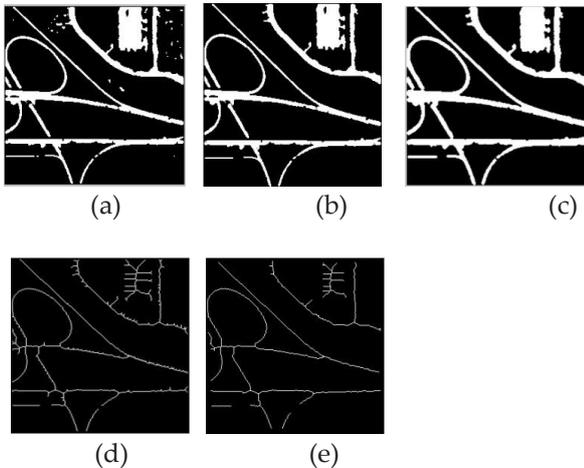


Fig. 4.3 a. Result of PCNN based segmentation b. Result after removal of small regions c,d. Result after applying morphological operations e. Extracted Roads

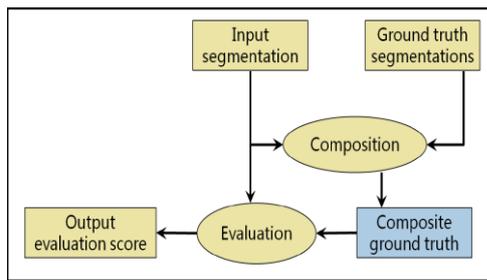


Fig.3.1 Flowchart of the segmentation evaluation framework.

TABLE 4.1

Method	Precision	Recall	Accuracy
Chan-Vese Active Contour Model	0.0350	0.1378	0.8647
PCNN	0.3802	0.4803	0.9712

Comparative Analysis of Performance of the Segmentation Techniques

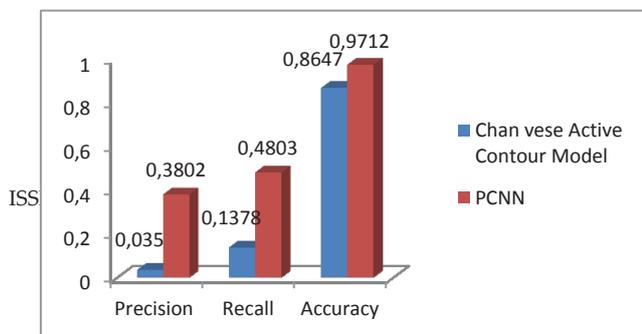


Fig. 4.1 Performance analysis of Chan-Vese active contour and PCNN methods

Notes

- /1/ Amini, J., Saradjiana, M. R., Blaisc, J., Lucas, C., & Azizi, A. (2002). Automatic road-side extraction from large scale image maps. *International Journal of Applied Earth Observation and Geoinformation*, 95–107.
- /2/ Gong, P., & Wang, J. (1997). Road Network Extraction from Airborne Digital Camera Images. *Geoscience and Remote Sensing, IGARSS* (pp. 895 - 897). Singapore: IEEE.
- /3/ Gagalowicz, A., (1987). *Texture Modelling Applications*. The Visual Computer. Springer-Verlag, Berlin.
- /4/ Mena, J., & Malpica, J. (2005). An automatic method for road extraction in rural and semi-urban areas starting from high resolution satellite imagery. *Pattern Recognition Letters*, 1201-1220.
- /5/ Mirnalinee, T., Das, S., & Varghese, K. (2009). Integration of Region and Edge-based information for Efficient Road Extraction from High Resolution Satellite Imagery. *Advances in Pattern Recognition* (pp. 373 - 376). Kolkata: IEEE.
- /6/ Hinz, S., & Baumgartner, A. (2003). Automatic extraction of urban road networks from multi-view aerial imagery. *ISPRS Journal of Photogrammetry & Remote Sensing*, 83-98.
- /7/ Gaetano, R., Zerubia, J., Scarpa, G., & Poggi, G. (2011). Morphological road segmentation in urban areas from high resolution satellite images. *Digital Signal Processing (DSP)* .pp. 1 - 8).
- /8/ D. Terzopoulos and K. Fleischer. (1988). Deformable models. *The Visual Computer*, 4(6):306-331.
- /9/ V. Caselle, F. Catte, T. Coll, and F.Dibos. (1993). A geometric model for active contours in image processing. 66:1-31.
- /10/ V.Caselles, R. Kimmel, and G.Sapiro. (1997). Geodesic active contours. *International Journal of Computer Vision*, 22(1):61-79.
- /11/ S. Kichenassamy, A. Kumar, P. Olver, A. Tannenbaum, and Z. Yezzi., (1995). Gradient Flows and geometric active contour models. *ICCV*.
- /12/ Stanley Osher and James A. Sethian. (1988). Fronts propagating with curvature dependent speed; algorithms based on hamilton-jacobi formulations. *J. Comput. Phys.*, 79:12-49.
- /13/ D. Mumford and J. Shah., (1989). Optimal approximation by piecewise smooth functions and associated variational problems. *Comm. Pure Appl. Math*, 42:577-685.
- /14/ T. Chan and L. Vese, (2001). Active Contours without edges. In *IEEE transactions on image processing* 10(2), pp. 266-277.

- /15/ Luminata Vese and Tony Chan. (2002). A multi-phase level set framework for image segmentation using the Mumford and Shah model. *International Journal of Computer Vision*, 50(3):271-293.
- /16/ T. Chan and L. Vese, (2001). Active Contours without edges. In *IEEE transactions on image processing* 10(2), pp. 266-277.
- /17/ J. M. Kinser, T. Lindbladh, *Image processing with pulse-coupled Neural Networks*. Springer- Verlag London Limited.
- /18/ M. Egmont-Petersena., D. de Ridderb, H. Handelse *Image Processing with Neural Networks-a review*.
- /19/ W. Maass. *Network of spiking Neuron -The Third generation of Neural Network model*, Neural Network.
- /20/ Hu, J., Razdan, A., Femiani, J., Ming, C., & Wonka, P. (2007). Road Network Extraction and Intersection Detection from Aerial Images by Tracking Road Footprints. *Geoscience and Remote Sensing, IEEE Transactions on*, 4144 - 4157.
- /21/ V.Caselles, R. Kimmel, and G.Sapiro. (1997). Geodesic active contours. *International Journal of Computer Vision*, 22(1):61-79.
- /22/ Martin, D. (2002). An empirical approach to grouping and segmentation. Ph.D. dissertation U. C. Berkeley
- /23/ Kevin McGuinness and Noel E. O'Connor. (2008). *A Comparative Evaluation of Interactive Segmentation Algorithms*. Elsevier.
- /24/ Manisha Sharma and Vandana Chouhan. (2012). Objective Evaluation Parameters of Image Segmentation Algorithms. *International Journal of Engineering and Advanced Technology (IJEAT)* ISSN: 2249 – 8958, Volume-2, Issue-2.
6. Cross, G.R., Jain, A.K., 1983. Markov random field texture models. *IEEE Trans. Pattern Anal. Machine Intell.* 5 (1).
7. De Angelis, G., Ohzawa, I., & Freeman, R. D. (1995). Receptive-field dynamics in the central visual pathways. *Trends Neuroscience*, 451–458.
8. Del-Toro-Almenares, A., Mihai, C., Vanhamel, I., & Sahli, H. (2007). Graph Cuts Approach to MRF Based Linear Feature Extraction in Satellite Images. *Progress in Pattern Recognition, Image Analysis and Applications*, 162-171.
9. Dubuisson, M.P., Gupta, J., Gupta, A., (1996). Color and texture fusion: Application to aerial image segmentation and GIS updating. In: *Proc. Third IEEE Workshop Applicat. Comput. Vision. WACV96*, pp. 2–7.
10. Grote, A., & Heipke, C. (2008). Road Extraction for The Update of Road Databases in Suburban Areas. *Archives of Photogrammetry, Remote Sensing and Spatial Information Science*, 563-568.
11. Grote, A., Heipke, C., Rottensteiner, F., & Meyer, H. (2009). Road extraction in suburban areas by region-based road sub graph extraction and evaluation. *Urban Remote Sensing Joint Event* .pp. 1 - 6. Shanghai: IEEE.
12. Guan, J., Wang, Z., & Yao, X. (2010). A new approach for road centerlines extraction and width estimation. *Signal Processing (ICSP)* .pp. 924 - 927. Beijing: IEEE.
13. Hauptfleisch, A. C. (2010). *Automatic Road Network Extraction from High Resolution Satellite Imagery using Spectral Classification Methods*. Pretoria: University of Pretoria.
14. Jin, H., Feng, Y., & Li , B. (2008). Road network extraction with new vectorization and pruning from high-resolution RS images. *Image and Vision Computing* (pp. 1 - 6). Christchurch: IEEE.
15. Kirthika, A., & Mookambiga, A. (2011). Automated Road Network Extraction Using Artificial Neural Network. *Recent Trends in Information Technology (ICRTIT)* (pp. 1061 - 1065). Chennai: IEEE.
16. Lee, H. Y., Park, W., Lee, H.-K., & Kim, T.-g. (2000). Towards Knowledge-Based Extraction of Roads from 1m-resolution Satellite Images. *Image Analysis and Interpretation, 2000. Proceedings. 4th IEEE Southwest Symposium* (pp. 171 - 176). Austin: IEEE.
17. Li, G., An, J., & Chen, C. (2011). Automatic Road Extraction from High-Resolution Remote Sensing Image Based on Bat Model and Mutual Information Matching. *Journal of Computers*, Vol. 6, No. 11, 2417-2426.
18. Lin, X., Zhang, J., Liu, Z., & Shen, J. (2008). Integration method of profile matching and template matching for road extraction from high resolution remotely sensed imagery. *Earth Observation and Remote Sensing Applications* (pp. 1 - 6). Beijing: IEEE.

Literature

1. Bajcsy, R., & Tavakoli, M. (1976). Computer Recognition of Roads from Satellite. *IEEE Transactions on Systems and Cybernetics*, Vol. SMC-6,, 623-637.
2. Bong, D. B., Lai, K. C., & Joseph, A. (2009). Automatic Road Network Recognition and Extraction for Urban Planning. *International Journal of Applied Science, Engineering and Technology*, 54-59.
3. Cheng, H.D., Jiang, X.H., Sun, Y., Wang, J., (2001). Color image segmentation: Advances and prospects. *Pattern Recognition* 34, 2259–2281.
4. Chen, J., Pappas, T.N., (2002). Adaptive image segmentation based on color and texture. In: *Proc. ICIP*, Rochester, NY.
5. Clark, A.A., Thomas, B.T., Campbell, N.W., Greenway, P., (1999). Texture deconvolution for the Fourier based analysis of non-rectangular regions. In: *Brit. Machine Vision Conf.*, Nottingham, UK, pp. 193–202.

19. Ma, H., Qin, Q., Du, S., Wang, L., & Jin, C. (2007). Road Extraction from ETM Panchromatic Image. Geoscience and Remote Sensing Symposium IGARSS 2007. IEEE International (pp. 460-463). Barcelona: IEEE International.
20. Mathiassen, J.R., Skavhaug, A., Bo, K., (2002). Texture similarity measure using Kullback–Leibler divergence between gamma distributions. In: Comput. Vision ECCV 2002, Copenhagen, Lecture Notes Computer Science, vol. 2352, 133 p.
21. Maurya, R., Gupta, P., & Shukla, A. S. (2011). Road Extraction Using K-Means Clustering and Morphological Operations. *Image Information Processing (ICIIP)* (pp. 1 - 6). Himachal Pradesh: IEEE.
22. Mayer, H., Laptev, I., & Baumgartner, A. (1998). Multi-scale and Snakes for Automatic Road Extraction. In H. Burkhardt, & B. Neumann, *Computer Vision ECCV* (pp. 720-733). Berlin: Springer.
23. Mena, J. (2003). State of the art on automatic road extraction. *Pattern Recognition Letters*, 24, 3037–3058.
24. Mirmehdi, M., Petrou, M., (2000). Segmentation of color textures. *IEEE Trans. Pattern Anal. Machine Intell.*
25. Péteri, R., & Ranchin , T. (2003). Multiresolution snakes for urban road extraction from ikonos and quickbird images. *23rd EARSeL Annual Symposium Remote Sensing in Transition*. pp. 2-5. Rotterdam: Millpress.
26. Trinder, J. (2009). Towards Automation of Information Extraction from Aerial and Satellite Images. In D. Li, J. Shan, & J. Gong, *Geospatial Technology for Earth Observation* (pp. 289-327). Berlin: Springer.
27. Van Cleijnenbreugel, J., Osinga, S.A., Fierens, F., Suetens, P., Oosterlinck, A., (1991). Road extraction from multi-temporal satellite images by an evidential reasoning approach. *Pattern Recognition Lett.* 12, 371–380.
28. Wiedemann, C., & Ebner, H. (2000). Automatic Completion and Evaluation of Road Networks. *International Archives of Photogrammetry and Remote Sensing*, 979-986.
29. Xiao, Z.-q., Bao, G.-s., & Jiang, X.-q. (2004). Road network extraction in classified SAR images using genetic algorithm. *Journal of Central South University of Technology*, 180-184.
30. Yanqing, W., Yuan, T., Tai, X., & Shu, L. (2006). Extraction of Main Urban Roads from High Resolution Satellite Images by Machine Learning. *7th Asian conference on Computer Vision* (pp. 236-245). Heidelberg: Springer.
31. Zhang, Q., & Couloigner, I. (2006). Benefit of the angular texture signature for the separation of parking lots and roads on high resolution multi-spectral imagery. *Pattern Recognition Letters*, 937-946.
32. Zhao, L., & Wang, X. (2010). Road extraction in high resolution remote sensing images based on mathematic morphology and snake model. *Image and Signal Processing (CISP)* (pp. 1436 - 1440). Yantai: IEEE.
33. Zhao, L., & Xili, W. (2010). Road Extraction in High Resolution Remote Sensing Images Based on Mathematic Morphology and Snake Model. *International Congress on Image and Signal Processing* (pp. 1436-1440). Yantai: IEEE.