AN INTEGRATED APPROACH TO REAL-TIME MULTISENSORY INSPECTION WITH AN APPLICATION TO FOOD PROCESSING

A Dissertation
Presented to
The Academic Faculty

by

Yuhua Ding

In Partial Fulfillment
of the Requirements for the Degree of
Doctor of Philosophy in the
School of Electrical and Computer Engineering

Georgia Institute of Technology
Atlanta, GA

December 2003

AN INTEGRATED APPROACH TO REAL-TIME MULTISENSORY INSPECTION WITH AN APPLICATION TO FOOD PROCESSING

Approved:

Dr. J. Lewis Dorrity

Dr. Magnus Egerstedt

Dr. Bonnie S. Heck-Ferri

Dr. George J. Vachtsevanos, Chairman

Dr. Douglas B. Williams

Dr. Anthony J. Yezzi, Jr

Date Approved by Chairman: November 24, 2003

DEDICATION

TO MY HUSBAND, XUDONG WANG,

AND MY SON, ANDREW YUANZHOU WANG,

WHO MADE THE JOURNEY ENJOYABLE

ACKNOWLEDGEMENTS

First of all, I would like to express my special and sincere gratitude to my advisor Dr. George Vachtsevanos for his persistent guidance and trust throughout my research. His enthusiasm for the career affected me greatly. I would also like to thank my co-advisor Dr. Bonnie Heck-Ferri and Wayne Daley, my project supervisor, for their support and technical discussions.

Secondly, I wish to thank Drs. Anthony Yezzi, Magnus Egerstedt, Douglas B. Williams, and J. Lewis Dorrity for serving my committee. I thank them for their time, support, and suggestions. I must thank Dr. Anthony Yezzi for spending much time to teach me and to discuss with me. His source code expedited my research progress considerably.

I thank all present and past members of the Intelligent Control Systems Laboratory, whose assistance, encouragement, and friendship made the lab a lovely place. Special thanks must be given to Yingchuan Zhang, Dr. Guangfan Zhang, Dr. Lichu Zhao, Dr. Liang Tang, and Dr. Biqing Wu, for their sincere friendship, wam-hearted and on-demand help to me and to my family, and technical and moral support throughout the research.

I am forever indebted to my parents, Xianglin Cheng and Dun Ding, for constantly providing me the moral support and encouragement, which enabled me to achieve every goal in my life. I thank my big sister, Yuqing Ding, for being my sole model, my teacher, my friend, and my caregiver throughout the years.

Finally, I am deeply grateful to my husband Xudong Wang, who shared with me the good times and supported me through the bad times with his love, encouragement, incredible understanding, great tolerance, and sacrifices. Special thanks to my son Zhouzhou, who brought me joy everyday and made even the hardships enjoyable.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

SUMMARY

Real-time inspection based on machine vision technologies is being widely used in quality control and cost reduction in a variety of application domains. The high demands on the inspection performance and low cost requirements make the algorithm design a challenging task that requires new and innovative methodologies in image processing and fusion. In this research, an integrated approach that combines novel image processing and fusion techniques is proposed for the efficient design of accurate and real-time machine vision-based inspection algorithms with an application to the food processing problem.

Firstly, a general methodology is introduced for effective detection of defects and foreign objects that possess certain spectral and shape features. The factors that affect performance metrics are analyzed, and a recursive segmentation and classification scheme is proposed in order to improve the segmentation accuracy. The developed methodology is applied to real-time fan bone detection in deboned poultry meat with a detection rate of 93% and a false alarm rate of 7% from a lab-scale testing on 280 samples.

Secondly, a novel snake-based algorithm is developed for the segmentation of vector-valued images. The snakes are driven by the weighted sum of the optimal forces derived from corresponding energy functionals in each image, where the weights are determined based on a novel metric that measures both local contrasts and noise powers in individual sensor images. This algorithm is effective in improving the segmentation accuracy when imagery from multiple sensors is available to the inspection system. The effectiveness of the developed algorithm is verified using (i) synthesized images (ii) real medical and aerial images and (iii) color and x-ray chicken breast images. The results further confirmed that the algorithm yields higher segmentation accuracy than monosensory methods and is able to accommodate a certain amount of registration error. This feature-level image fusion technique can be combined with pixel- and decision-level techniques to improve the overall inspection system performance.

This research has lead to the following major contributions:

(a) Proposed a general methodology for real-time vision-based inspection of abnormality that contains certain shape and spectral patterns.

(b) Developed and tested a recursive segmentation and classification scheme for real-time segmentation with high accuracy.

(c) Applied the developed inspection methodology successfully to fan bone detection of deboned poultry meat and achieved satisfactory results.

(d) Derived and verified a fast snake-based multisensory image segmentation algorithm and applied it to the segmentation of visible band and x-ray images of deboned meat.

CHAPTER 1

INTRODUCTION

With advances in sensor techniques and computing power, machine vision techniques are becoming more efficient in detecting the presence of undesirable objects in a wide variety of products. For example, metal detection, optical, and x-ray inspection techniques are already available commercially ([18]). The problem of machine vision-based inspection involves target object detection and recognition using image processing and pattern recognition techniques. When the target objects possess certain patterns of spectrum, size, and shape, advanced image processing and analysis techniques will be necessary in detecting and identifying these objects with high accuracy. However, due to the large volume of the products being inspected, the processing time allowed for each image is extremely short. Many algorithms are disqualified by this constraint and the inspection system design becomes a demanding task. The first motivation of this research is to develop a general methodology for real-time inspection using fast and effective image processing and analysis techniques.

Inspection algorithms based on monosensory images may not produce satisfactory performance because of the ambiguity and incompleteness involved in the information. Computer vision systems that utilizes multispectral and hyperspectral imaging techniques have emerged as a more powerful solution to earth remote sensing, medical diagnosis, and agricultural applications [10]. Images acquired by different types of sensors are generally partially redundant and partially complementary. Fusion with redundant data can help reduce imprecision, and fusion with complementary data can provide a more complete description of the scene. In both cases, the inspection performance should be better than the monosensory performance. The fusion of multisensory information is not an easy task, though. Among all levels at which the fusion can be performed — pixel, feature, and decision — feature-level is well suited for performance improvement of real-time inspection, considering the amount of data involved and the details retained. Multisensory image segmentation is a type of feature-level fusion technique dealing with the segmentation of images generated from different sensors. As a replacement of their monosensory counterpart, the multisensory algorithms utilize the complementary information in multiple modalities to provide more accurate segmentation results. It makes smart decisions to eliminate redundancy and to resolve information conflict so that precise and meaningful interpretation of the segmentation results is possible. In this dissertation, a multisensory image segmentation algorithm will be developed based on the snakes. It will be tested on both synthesized and real images and the results will be compared to those of monosensory segmentation algorithms.

The application of bone detection in deboned meat is motivated by the dramatic increase in the demand for deboned meat over the last few years. The end users of the deboned product, especially the fast food corporations, are requesting zero bones in meat [35]. To address the problem of automated bone inspection in deboned meat, x-ray-based equipment has been used as a viable but expensive means. An x-ray inspection system works well in detecting large, embedded bones such as pulley bones. Feedback from customers indicates false alarm rates anywhere from 3-4% to 12-13% and detection rates of 95%-98% for these types of bones. However, such equipment has difficulty detecting thin surface bones, such as fan bones, a type of surface bone with a fan-like shape. Fan bones are typically less dense in calcium and of lower thickness perpendicular to the x-ray beam than the other bones normally found in the meat. Since the x-ray machine is optimized to detect embedded bones, it is difficult to detect fan bones using the same settings. Currently available systems detect fan bones with an accuracy of 30%-50%. The proposed inspection and multisensory segmentation techniques will be applied to the visible band and x-ray images of the deboned meat so that high inspection performance is achieved.

This dissertation is organized as follows. In the next chapter, the state-of-the-art techniques for vision-based real-time inspection are briefly reviewed, and a general methodology is developed for inspection of defects and foreign objects with certain shape and spectral patterns. The application of this methodology to the bone inspection problem is discussed in Chapter 3. In Chapter 4, an algorithm is developed for multisensory image segmentation based on snakes and results are presented on both synthesized and real images as well as color and x-ray images of deboned meat. Finally, the research is summarized and the contributions are listed in the last chapter.

CHAPTER 2

A GENERAL METHODOLOGY FOR VISION-BASED REAL-TIME INSPECTION

2.1 Real-Time Vision-Based Inspection

The problem of real-time machine vision (MV)-based inspection involves the detection and identification of foreign objects and defects using a variety of techniques including image processing and pattern recognition. It covers a wide range of research topics, with some shown in Figure 1. Each topic is critical to the performance of the inspection system. The focus of this research is mainly image processing, pattern recognition, and image fusion.

There exist potentially a large number of possible foreign objects and defects in the inspection process, such as metal, wood, glass, stone, scars, blemishes, etc., which are natural objects that do not follow specifications in size and shape. In some cases, they possess some pattern in shape, size, and color ([13]), while in others, they do not ([27]). On a specific production line, the types of foreign objects and defects are normally predictable. However, due to the speed requirement for real-time inspections, many image processing techniques become unsuitable. In this research, we are particularly interested in detecting, at a fast speed, foreign objects and defects that have consistent shape and spectral patterns with reasonable variability. An example of such objects is the fan bones in deboned poultry meat, which normally have a triangular shape and a dark red color. However, depending on the relative position to the camera, the fan bone features in the image exhibit a considerable variability in color and shape. Textural information is important in foreign object and defect detection in many applications, such as metal detection in free-flow products and some packaged foods (e.g., peas, peanuts, sweat corn, etc. [12]). Although for simplicity, we will not consider textural patterns in this research, the general methodology can be extended to include these features.

In this section, the requirements and common algorithms for vision-based inspection will be reviewed.

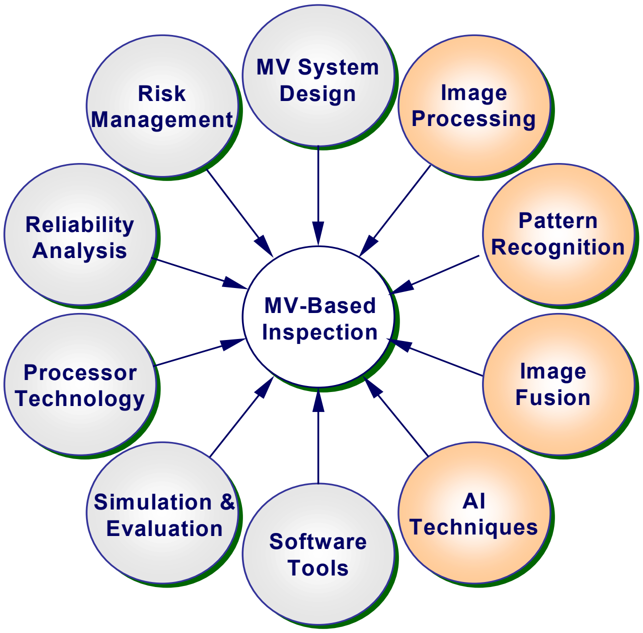


Figure 1. Some research topics in machine vision-based inspection.

2.1.1 Requirements of Real-Time Vision-Based Inspection

The vision-based inspection problem presents major challenges due to pattern complexity, the variability in defects/foreign objects and product samples, modifications on the inspection system, and the uncertainties on the product line. Defects and foreign objects may possess complicated patterns including color, contrast, texture, and location. Apart from the variability in the product properties, both the type and the pattern of the defects can vary as results of the alterations in the production process. Modifications on the inspection system components and drifting of the imaging conditions cause inconsistency among captured images. Finally, the uncertain aspects on the inspection line can be unpredictable. For example, the product may be positioned in an appropriate way that the target object is occluded and thus invisible to the imaging sensor.

All these factors impose strict requirements on inspection algorithms. Both the hardware and the software algorithms need to be carefully designed and verified in order to yield satisfactory, stable, and reliable performance.

For evaluation of the inspection performance, some most commonly used measurements and their definitions are discussed as follows.

- *Detectable defects*: The types of defects the system can recognize. For different defects, the listed metrics may vary.
- *Minimum size of detectable defect*: The minimum size of the defect the system can detect.
- *Accuracy*: The fraction of the objects that are correctly identified by the inspection system.
- *Detection rate* (*DR*): The fraction of the abnormal objects that are detected and correctly identified as abnormal by the inspection system.
- *False negative rate* (*FNR*): The fraction of the abnormal objects that are detected and incorrectly identified as normal by the inspection system. Note that *FNR*+*DR*=1.

- *False alarm rate* (*FAR*): The fraction of the normal objects that are incorrectly identified as abnormal by the inspection system.
- *Inspection speed*: The number of images that can be processed within a time unit (frames/minute).
- *Cost*: The total cost of the inspection system, energy consumption (electricity, water, etc.) and maintenance expenditures.

The false alarms in the inspection system result in extra load on processing and inspection and thus higher production cost. A bearable false alarm rate indicates the upper bound of *FAR* that the manufacturer can withstand. Sometimes the *DR* must be compromised somehow so that the *FAR* does not exceed the bearable *FAR*. Since *FAR* and *DR* are the combined results of all functional modules in an inspection algorithm, it is important to distribute the *FAR* and *DR* reasonably among modules so that the *DR* is maximized and the *FAR* is within the bearable *FAR* range.

The inspection speed must match or exceed the processing speed of the production line. When 100% inspection is required, the time window for each part to be inspected is very small, which makes most image processing algorithms unusable.

In addition to the above measurements, an inspection system must have certain attributes in order to meet the requirements of the manufacturing requirements, for example, low sensitivity, reliability, robustness, flexibility, and human interface for service people and operators. Unfortunately, these attributes are not only hard to quantify, but also ambiguous and imprecise. For example, reliability can mean that the performance is not greatly affected by either of the following situations: the camera is shaken loose or the computer momentarily cannot keep up with the product flow because of excessive analysis time on one image.

As has been pointed out in [11], all performance measurements will be interlinked, corresponding to a constraining surface in the multidimensional space, so that adjusting one measurement forces the adjustment of others. In addition, each inspection algorithm will have its own constraining surfaces due to specifications and manufacturing conditions.

Considering the diversity of applications and intrinsic complexity of the inspection problem, it is impossible to claim an algorithm to have the optimal performance. Instead, we focus on developing an effective methodology with satisfactory performance on the detection of defects and foreign objects with certain spectral and shape patterns.

2.1.2 Review of Vision-Based Inspection Algorithms

A review of the literature on the inspection systems revealed that MV-based inspection algorithms usually consist of three modules: image acquisition, image processing, and decision making, as illustrated in Figure 2. Three example vision-based inspection systems are summarized in Table 1 and compared in terms of the algorithms adopted and the system performance. Some common techniques and algorithms used in each of the three module are briefly described below.

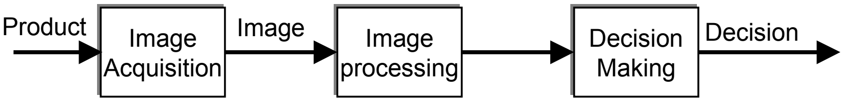


Figure 2. General methodology for vision-based inspection.

Table 1. Comparison of three vision-based inspection systems.

System		***Patel, 1994*** [27]	***Wang, 2000*** [45]	***Boyer, 2001*** [2]
Application		Foreign objects in sealed food (corn, peas, etc.)	Wineglass defect	Pipeline corrosion
Sensor		X-ray	Visual (grayscale)	Range
ALGORITHM	***Processing***	(a) Log-transform (b) Thresholding (c) Adaptive thresholding (d) Texture analysis	(a) Gabor filter	(a) Finite window robust sequential estimator (b) Thresholding (c) Cluster
	Features	/	Contrast, local homogeneity, entropy, etc.	Depth, area, centroid, aspect ratio
	Classification	Pixel counter	BPNN	/
Detection Rate		Stone-100% Metal-95% Glass-80% Others (soft) - <60%	97.5%	/
Number of test samples		125 images	/	/
Speed		/	/	1sec on 300x300 clip in preprocessing
Comments		No *a priori* knowledge about products or defects	Parameters in Gabor filter	Not an automated inspection system

2.1.2.1 Image Acquisition

Imagery for inspection purposes can be generated from a variety of sensing techniques, such as visible, infrared (IR), ultraviolet (UV), laser, and x-ray sensors. Visible, IR, and UV sensors measure the intensity of the light emitted or reflected by the object within the sensitive spectral range of the sensor. X-ray sensors measure the permeability and thickness of the object. Laser sensor utilizes structured light and generates range data that corresponds to the distance from the object to the sensor. Currently, although 3-D images are available, 2-D images are mostly used, especially in industrial applications, because of the complexity and data volume involved in 3-D image acquisition, visualization, and processing.

An imaging system usually consists of the following hardware components:

- *Camera* generates the imagery of the scene. A single sensor or multiple sensors are housed in the camera to generate image data. Camera settings include shutter speed, iris control, trigger mode, brightness, zoom, white balance, frame rate, and so on. Some cameras have built-in processors for low-level image processing. Recently, cameras with the frame rates of 75 - 8000 frames/sec have been available using CMOS-sensors, which urgently calls for extremely fast real-time image processing techniques.
- *Data acquisition board* (*frame grabber*) collects the data generated by the camera, forms the image, then transfers the image to the data storage unit of the processor. Some products also have on-board programmable image processing capabilities to reduce the preprocessing load and the traffic of the host computer and also to increase the processing speed on basic image operations such as convolution.
- *Illuminator* is designed according to the sensor requirement and scene characteristics. The illuminator is calibrated to maximize the contrast both between the product and background (such as a conveyor belt) and between the abnormal and the normal objects, while keeping the artifacts at the minimum level.

Current inspection systems mostly rely on a single type of sensor to generate the imagery data. However, because of the limitation of the sensor characteristics, the location of the sensor, the setup of the imaging system, or the variability of the products, monosensory data can be too ambiguous to interpret. Each type of sensor has its distinctive characteristics. When different types of sensors are used at the same time, the acquired images provide complementary information, which can be utilized to resolve the ambiguity involved in monosensory data. In recent years, multispectral and hyperspectral imaging has emerged as a powerful technique in remote sensing, medical imaging, and industrial and agricultural applications [10]. Because of the complexity involved in multisensory imagery, image registration and fusion techniques are required in addition to advanced image processing algorithms in order to process and interpret the data.

2.1.2.2 Image Processing

The image processing module usually consists of a series of operations. The objective of these operations is to identify the possible positions of the potential abnormality. Various filtering and thresholding algorithms are adopted to enhance the image contrast and also to remove irrelevant information. A segmentation step may be needed to further isolate the regions of abnormality for identification in the decision module. To solve the segmentation problem, sometimes a thresholding technique suffices, as in metal detection in food [27], where the difference in intensity and texture patterns between the metal objects and the food stuff is prominent. However, when the difference is inconspicuous or variable, a thresholding technique alone is not able to produce reliable performance. In this case, more sophisticated algorithms are necessary, such as adaptive thresholding, morphological filtering, snake-based segmentation, or neural-net-based segmentation. Unfortunately, many of these segmentation algorithms are time consuming and thus inappropriate for real-time applications. Even for the faster ones, extreme caution must be taken in their implementation. Usually the processing time can be reduced considerably by incorporating the *a priori* knowledge of the product and the potential abnormality.

Since this research is particularly interested in detecting abnormality with shape patterns, we will briefly review the segmentation algorithms below with a special interest in techniques that can improve the accuracy.

2.1.2.2.1 Segmentation Features and Image Segmentation Algorithms

Image segmentation is the process to map an image $I(x, y)$ over a spatial domain Ω to a labeled image $J(x, y)$. In $J(x, y)$, each group of connected pixels with the same label is called a *region* (R), which is separated from each other by *edges*. The segmentation process can be viewed as approximating image I using the piece-wise constant cartoon image J which has only a limited number of intensity levels. In this research, we consider the simplest case, where J has only two possible labels. This corresponds to separating the target objects from the background. In the inspection problem, the target object is the foreign object or the defect, while the background can be the conveyor belt or the normal product.

The segmentation features of an image include line segments, points of interest, edges, and regions. They may be viewed as an elementary or primitive attribute associated with an image or a region in the image. Note that segmentation feature is a different concept from the numerical features that describe the characteristics of a segmented region.

Although edge and region are said to be “dual,” they refer to two different image properties: local differences and global homogeneity [1], and their behaviors are different. In order to be segmented successfully, the image intensity needs to change smoothly within each region while abruptly from one region to another. However, it is not always the case in real images due to the following reasons, which makes the segmentation a demanding task.

- The image exhibits various features resulted from the reflectance of objects in the scene, illumination, angle of view, and facets of objects. It is possible that two distinct objects cannot be distinguished from each other in the image.
- The artifacts, such as shadows and glares, often cause over-segmentation errors, where an object is divided into several regions, and edge features are observed between neighboring regions.
- Images can be noisy.

Segmentation algorithms can be classified into two categories: edge-based and region-based. Edge-based algorithms locate edges where the local difference is prominent, while region-based algorithms search for homogeneous regions according to some predefined criteria. The former generates an edge map (a binary image) with edge points highlighted, while the latter produces a gray-scale image of labels (region map). Since regions are usually used in scene interpretation, edge-based algorithms often employ a postprocessing step to retrieve regions from the edge map. Unfortunately, this step is both complicated and time consuming because edge chaining is necessary to connect the edge segments into closed curves. It is much easier to convert a region map to an edge map by obtaining the contours / boundaries of regions.

“Snakes”, or active contours, initially proposed by Kass *et al*. [24], is based on deforming an initial contour or surface to optimize an energy functional whose (local) minimum is obtained at the boundary of the desired features. An energy functional needs to be designed so that its (local) minimum is obtained at the boundary of the desired features. The energy usually consists of a term that controls the smoothness of the deforming curve and another one that attracts it to the boundary. The snakes is a widely adopted method and has been proven effective in tackling the segmentation of a variety of real images.

There are two types of segmentation models: region-based and edge-based. In the region-based models, statistics of entire regions (such as sample mean and variance) are used to direct the movement of the curve toward the boundaries of the features. This is in sharp contrast to the edge-based models, where the evolution of the curve depends strictly on nearby pixel intensities (*i.e.*, gradient information). Consequently, region-based models are global and normally do not need the inflationary terms, which are commonly seen in edge-based models, to drive the curve toward the feature boundaries. Region-based models are also more robust to noise since they do not employ differential operators, which are inherently sensitive to noise [42].

2.1.2.2.2 Accurate Segmentation Algorithms for Real-Time Applications

The performance of the segmentation algorithm is critical in image understanding, as will be seen from the example in Section 3.2. Various performance measures ([5][9][39][41]) have been proposed in the literature and a number of techniques have been developed to improve the segmentation performance utilizing these performance measures. For example, in [41], possible segmentation results are searched using simulated annealing for the optimal solution in terms of the specified performance. Unfortunately, this type of method can be very slow. Another common method is that after obtaining multiple segmentation results, the "best" is selected ([9][39]), or they are combined ([15][22]) to get a superior solution, according to the adopted performance measure. For example, in [9], the possible segmentations of an image are stocked in a tree called the segmentation tree. A recursive segmentation scheme is applied to the image using different segmentation operators until no unrecognized regions remain. The segmentations are compared using a cost function and thus each node of the tree is the optimal solution of the corresponding interest area. In [22], it is proposed to take the best of each segmentation method by performing competition. The obtained regions are interpreted using a multiscale fuzzy classifier, and then, the interpreted images are merged using different fusion operators. Methods of this type usually involve multiple times of segmentation, and therefore are not good candidates for real-time applications.

2.1.2.3 Decision Making

The decision making stage for real-time inspections can be as simple as a pixel counter, which indicates whether an image has been marked as having target objects by the segmentation algorithm ([27]). The neighborhood information can be combined to further eliminate spurious noises. More often, the decision stage contains a feature extraction module and a classification module. Various numerical features are first computed by the feature extraction module based on the region map obtained from the image processing module; then, a classifier is deployed to claim if an abnormal object is present. Candidate features include the following:

- *Spectral features*. Spectral features are computed from the intensity distributions in all channels within a region. Both mono and multichannel features can be computed, such as mean, variance, and higher moments of the intensity, hue, or saturation.
- *Shape features*. Shape features are usually expressed in terms of the pixel dimension. Normally used shape features include perimeter, area, centroid, major and minor axes, thickness, hole-based shape features, statistical moments, symmetry, shape signatures, topological descriptors, and so forth ([6]).
- *Other features*. Features other than spectrum and shape can be derived from the *a priori* information about the abnormality and the product, such as the position of the abnormality.

Most numeric features proposed in the literature have good performance only in specific situations because of the degenerated mapping from 2-D data set to 1-D features. Nevertheless, even some simple choices of features can be applied successfully to specific situations. Feature selection is the process to choose a set of features that can separate classes effectively. For many problems, it is a demanding task. A common approach is to test a large set of candidate features and then apply an automatic feature selection algorithm to define a proper set of features with respect to a given training set.

Classification algorithms are usually employed to assign a class label to each considered region. There are two particularly important aspects related to object classification. The first is the problem of deciding whether an input feature vector belongs to some specific predefined class, which is usually known as *supervised classificatio*n. The second aspect is how to define or identify the involved classes in a population of previously unclassified objects. This represents a difficult task, and expert knowledge acquisition problems are usually involved. The latter situation is known as *unsupervised classification* or *clusterin*g. Since the products and type of abnormality being detected are usually known when designing an inspection algorithm, supervised classification is usually adopted because of its higher accuracy. However, sufficient sample data need to be collected to train the classifier, which is not only time consuming, but also sometimes unachievable due to the limitation from the production process.

2.2 A General Methodology for Vision-Based Inspection

The proposed vision-based inspection methodology consists of an on-line image acquisition setting and a software detector tool that is optimized for the real-time inspection task. A diagram illustrating this scheme is shown in Figure 3.

The proposed scheme has five stages: image acquisition, preprocessing, snake-based segmentation, feature extraction, and classification using a probabilistic neural network classifier. First, multispectral high-resolution images are acquired on-line using the high-performance imaging system shown in Figure 4, in which a dome-shaped illuminator is designed to produce uniform diffusive lighting so that the object-background contrast is maximized with minimal artifacts. The acquired images are then preprocessed using a series of image processing techniques to further enhance the contrast and to remove irrelevant information. The adapted segmentation algorithm is the snakes algorithm that is optimized for maximum speed. The energy functional of the snakes is constructed based on the image characteristics. From the segmentation results, spectral and shape features are extracted for each region. Finally, classification is performed using a probabilistic neural network (PNN) classifier to obtain the probability and, when desired, the confidence level that the input feature belongs to each class. The class with the highest probability is claimed to be the feature's identity. However, if the confidence level is not high enough, the segmentation can be repeated to improve the segmentation accuracy, which usually increases both the classification confidence and the overall inspection accuracy.

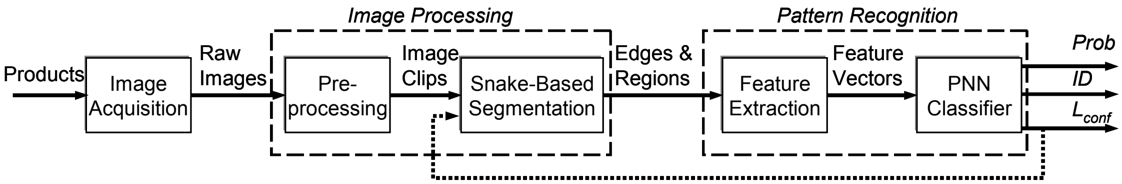


Figure 3. A general methodology for real-time vision-based inspection.

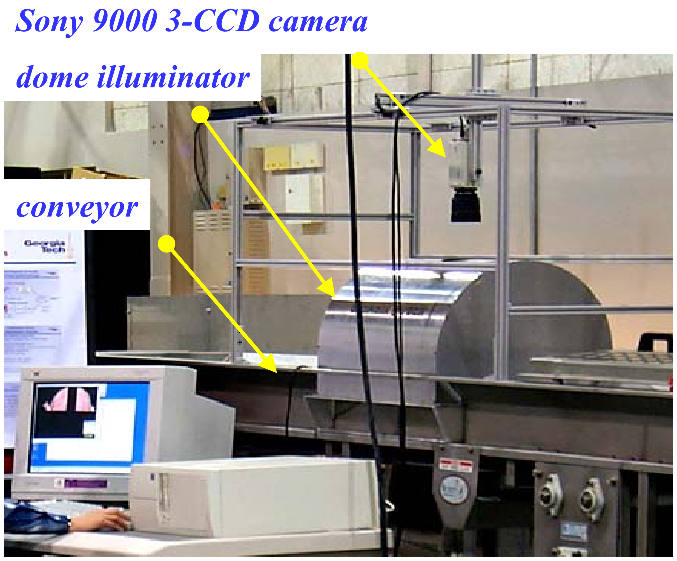


Figure 4. The image acquisition system hardware setup.

This scheme is different from existing real-time vision inspection algorithms in terms of the following aspects:

- The cloudy-day illuminator is designed to produce high-quality image data. It houses various light sources with linear profiles.

- The segmentation algorithm adapted is the active contour method implemented using the level set methods. It is highly optimized for real-time applications.
- A recursive segmentation and classification strategy is developed to improve the segmentation accuracy and in turn the overall system performance.
- Mulsisensory imaging techniques can be easily incorporated into this scheme using the multisensory segmentation method proposed in Chapter 4.

This methodology is in general applicable to any vision-based object detection problem, especially to the real-time inspection of foreign objects or defects whose type and characteristics are already known beforehand.

In this section, we will first present a method for performance estimation when designing inspection algorithms, then each module is separately discussed as a guideline for applying the proposed methodology to a specific application.

2.3 Performance Estimation and Individual Module Design

The design of inspection algorithms can be formulated as an optimization problem: choose a set of algorithms $\{A_i\}$ $i=1, \ldots, M$, and their parameters to optimize the objective function E that is expressed in the performance measures specified in Section 2.1.1, while at the same time satisfying the constraints in system specifications:

$$\max_{A_i \in \text{available algorithms}} \{ \max_{\text{parameters in } A_i} \{E(FA, DR, \ldots)\}\}, \quad \text{s.t.}\, condition_1, \ldots, condition_N, i = 1, \ldots, M .$$

For each inspection problem, the procedure shown in Figure 5 needs to be followed to optimize the system performance ([11]).

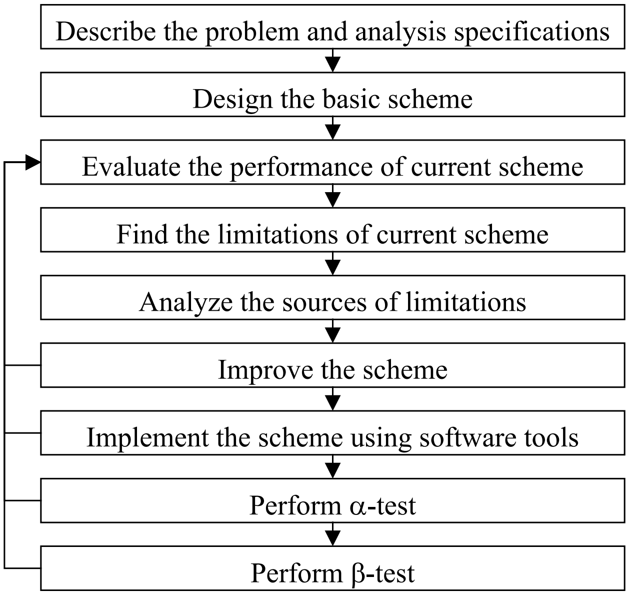

Describe the problem and analysis specifications
Design the basic scheme
Evaluate the performance of current scheme
Find the limitations of current scheme
Analyze the sources of limitations
Improve the scheme
Implement the scheme using software tools
Perform α-test
Perform β-test

Figure 5. Inspection algorithm design procedure.

The design process involves multiple cycles of algorithm design, implement, evaluation, and improvement. Generally, this is a very time consuming and tedious task. For our scheme that involves multiple stages, this process becomes even more complicated since even a minor change in an earlier stage affects the performance of the following modules. Apparently, some general guidelines are necessary in designing and implementing such inspection algorithms in order to shorten the development cycle efficiently with satisfactory and stable performance. Moreover, it would be very helpful if the functional modules can be decoupled in some way so that each one can be designed and evaluated independently.

2.3.1 Estimation of Performance Metrics

Each stage in the general flow chart may contain more than one functional module. They can be arranged either in series or in parallel. Thus, the overall performance depends on not only the performance of all modules, but also their arrangement.

For each functional module, the performance metrics listed in Section 2.1.1 can be calculated, as illustrated in Figure 6. Each module claims a certain number of inputs as “normal” or “abnormal” and passes the remaining to its successive module. Thus, each module generates a certain number of *FN*'s and *FA*'s. For example, for a module that performs filtering, if the contrast between the abnormal feature and its surrounding pixels is eliminated during the process, an *FN* is generated. The effect of these errors on the overall system performance, though, also depends on the arrangement of modules.

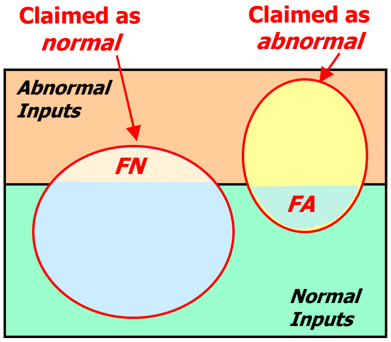


Figure 6. The performance measures of a functional module.

When two modules are in series, the former module acts as a filtering module for its successor. Since the *FN*'s and *FA*'s claimed in the previous module will not be passed to the next one, both errors accumulate with this arrangement. An example is the processing inside a region of interest (ROI). In case the defect falls out of the ROI, it will not be detected and a *FN* generates.

When two modules are in parallel, the performance estimation is more complicated because the decision will be made in a following fusion center as for how to combine the outputs from these two modules. In this case, the fusion rules together with the performance of each module determine the final performance. An example is shown in Figure 7, where the input image is fed into two filters, each generating a binarized image: B_1 and B_2, respectively. The following fusion center determines how B_1 and B_2 are combined into one binary image B. An example scenario is depicted in Figure 8 showing B_1 and B_2 as well as the combined binary images resulted from pixel-wise logical *AND* and *OR*, respectively. It can be seen from this example that in order to estimate the final errors from individual module performance, the failure pattern of each module must be analyzed so that a rule that generates minimal errors can be selected.

It is necessary to point out that the performance of each module may vary dramatically because of the limitation of the adopted algorithm and the input data's characteristics. A minor change in the previous module may cause significant performance change in its successive module, which dramatically increases the complexity of the inspection algorithm design.

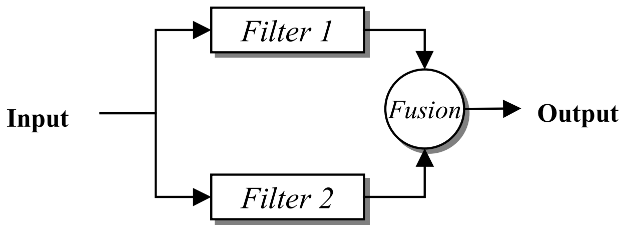


Figure 7. Two filters in parallel followed by a fusion center.

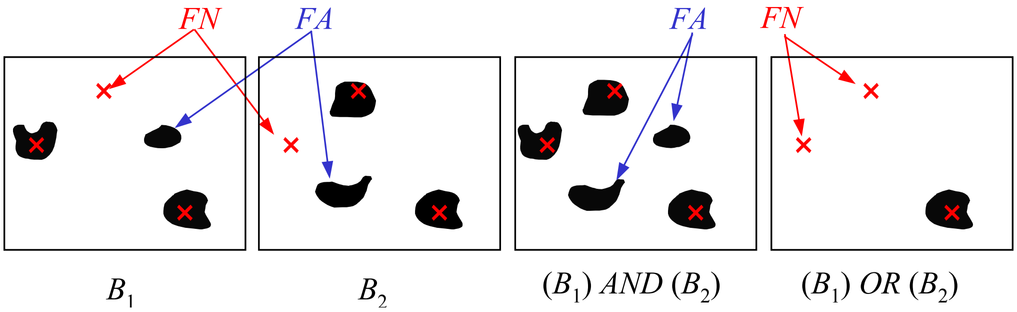


Figure 8. An example of the output binary images using different fusion rules. Red crosses indicate the positions of the abnormality. Dark pixels are the detected abnormality.

2.3.2 Individual Module Design

In the proposed method, all stages are concatenated, which significantly eases the performance estimation since each stage can be designed and assessed independently before they are put together for final assessment. Nevertheless, extreme caution must be taken when claiming an image as "normal" or "abnormal" at an intermediate stage, since all errors add directly to the final performance.

Specifically, the following factors plays the most important roles and special attention needs to be paid in designing the corresponding stages:

- *FNR* and *FAR* in preprocessing
- Segmentation accuracy
- Feature selection
- Classifier performance

Although the choice of specific algorithms at each stage is application specific, there are some common issues involved in applying the proposed methodology. In the remaining of this section, each stage will be discussed in detail as for how to choose appropriate algorithms in order to achieve satisfactory performance.

2.3.2.1 Preprocessing

The first goal of preprocessing is to remove irrelevant information using *a priori* knowledge of the image. This reduces the amount of data in later processing stages and increases the processing speed. However, caution must be taken in data removal since the data is not recoverable once removed. For example, in the fan bone detection application, assuming that fan bones are removed by mistake in preprocessing from 1% bone-contaminated sample images, they will not have the chance to be detected by later modules, and thus the final *DR* will not exceed 99%. In addition, the algorithm needs to be so simple that its total processing time is less than that in processing the excessive data by later steps.

The second goal of this step is to obtain a reasonable estimate of the target for segmentation. Since most segmentation algorithms are local in nature, the initial estimate affects greatly the final accuracy. By exploring the possible locations of suspected abnormality, we are able to focus the segmentation in the neighborhood of these positions. The advantages include increased segmentation speed with generally more accurate segmentation results. Again one must be cautious since a target that is missed at this point will be most probably unrecoverable by the segmentation algorithm.

Depending on the image quality and the object-background contrast, operations such as denoising and histogram equalization may be necessary in order to improve the detectibility of the defects and foreign objects.

Simple and fast algorithms such as thresholding and filtering are usually effective and sufficient in this stage.

2.3.2.2 Segmentation Using Active Contour Methods

Normally speaking, active contour methods are slow and not suitable for real-time applications. The snake adapted in this research, however, is a highly optimized version developed for real-time applications. It is implemented using the level set methods proposed by Sethian ([33]) in order to handle topological changes of contours such as merging and splitting. An ultra-narrow band scheme is adopted to further reduce the computation load by updating the level set values at only the nearest neighbors of the contour points[1]. Moreover, the techniques that explore the *a priori* knowledge about the samples and the abnormalities, including specification of the region of interest, estimation of the initial contour, and elimination of the termination criteria, can be utilized to further increase the speed and accuracy.

Segmentation accuracy has a significant effect on feature distributions and classification performance. The example of fan bone detection given in Section 3.2 shows the relationship between the segmentation accuracy and the classification performance. To improve the overall classification accuracy, more accurate segmentation schemes are needed. However, because of the complexity of the problem, it is hard to find a single segmentation routine that meets the accuracy requirement. To solve this problem, different segmentation algorithms can be applied and the results are then combined to produce higher accuracy, as have been reviewed in Section 2.1.2.2.2. However, such

---

[1] The ultra-narrow band method was developed by Dr. A. Yezzi, who is a faculty member in School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA.

techniques have slow processing speed and thus are unsuitable for real-time applications. Here we propose a faster recursive segmentation and classification method to achieve accurate segmentation at a much lower computation cost. The basic idea of this method is to repeat the segmentation only when the results are found to be inaccurate by the classifier. The concept of classification confidence is utilized to assess the segmentation accuracy. The proposed scheme of recursive segmentation and classification is shown in Figure 9.

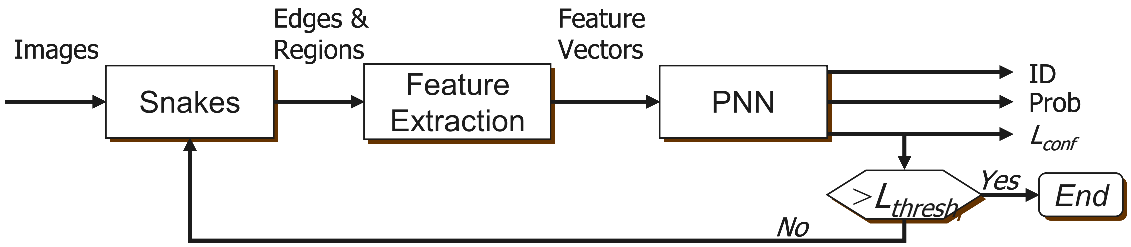


Figure 9. **Recursive segmentation and classification.**

Using the concept of confidence level, a label k is assigned to a region only when the confidence level L_k of the classifier about this label is higher than a positive threshold. Otherwise, the region is claimed as not sure or *unclassified*, and segmentation needs to be repeated, using the same algorithm with a different set of parameters or a different algorithm, until a result with desired confidence level is achieved.

The measurements of *absolute accuracy* (a), *relative accuracy* (A), and *repetition ratio* (r) are defined for the classifier as follows. Assume that of a total of M regions, m_c regions are classified correctly, m_e are classified incorrectly, and the remaining m_u are left unclassified, *i.e.*, $M = m_c + m_e + m_u$. Then we have

$$\begin{aligned} a &= m_c / M, \\ A &= m_c / (m_c + m_e), \\ r &= m_u / M, \end{aligned} \tag{1}$$

and the following relationship is easily established:

$$a = A(1 - r). \tag{2}$$

Appropriate segmentation and classification algorithms in each iteration are selected according to the characteristics of the images (especially of those with a poor segmentation accuracy from previous iteration).

The final accuracy and the total computation cost for segmentation and classification can be calculated as follows. Assume that the absolute accuracy *a* and repetition rate *r* are the same in each iteration. Then, after *n* rounds of segmentation and classification, the final accuracy becomes

$$accuracy = a\frac{1 - r^n}{1 - r} = A(1 - r^n) \tag{3}$$

and the relative computation time with regard to that on a single iteration is

$$\begin{aligned} relative\ computation\ time &= \frac{\text{computation time on } n \text{ iterations}}{\text{computation time on the first iteration}} \\ &= 1 + r + \cdots + r^{n-1} \\ &= \frac{1 - r^n}{1 - r} \end{aligned} . \tag{4}$$

As *n* approaches infinite, the final accuracy approaches its upper boundary of *a*/(1-r), which equals the relative accuracy *A*, and the segmentation cost approaches 1/(1-*r*) times of that on a single iteration scheme. The final accuracy can be improved through increasing *n* and *A*, both at the cost of higher computation expense. This is obvious with the number of iterations *n*. Now let us take a closer look at the relative accuracy *A*. Theoretically, *A* can be made closer to 1 by leaving more regions unclassified, *i.e.*, by decreasing m_c and increasing m_u. However, this results in lower a and higher r, and in turn more computation load. As a result, a trade-off must be made between the final accuracy and the total cost. Appropriate algorithms must be chosen in order to achieve high accuracy while keeping the computation overload tolerable.

2.3.2.3 Feature Extraction and Selection

Candidate features include spectral features, shape features, and other features such as location and adjacency. Shape features can be computed from either a region or the boundary of the region. Boundary-based features include curvature, perimeter, and Fourier descriptors, and region-based ones include aspect ratio, area, and moments. Not all features are suitable for real-time inspection because of the time constraint and product variability. For example, the boundary-based shape features computationally expensive in nature and therefore, should be avoided. Yet some other features, on the other hand, are highly sensitive to the variability in product conditions. An example is the mean intensity, which is largely determined by the lighting conditions. This type of features deteriorates the system reliability and increases design difficulty as well. Particularly, the following factors must be taken into account when selecting appropriate features:

- *Small overlapping of histograms between different classes*. Classes must be distinguishable using the selected features. A common method to assess the distinguishability is to use the *Fischer's Discriminant Ratio*.
- *Robustness to segmentation inaccuracy*. The segmentation algorithm does not always catch the target boundary accurately. Occasionally, adjusting parameters in the segmentation algorithm is useful in eliminating the inaccuracy. However, more

often, the result is unsatisfactory whatever parameters are chosen because of the lack of salient features in the image itself. The shape features, especially boundary-based features, are extremely sensitive to segmentation inaccuracy. Consequently, region-based shape features are preferred than boundary-based ones.

- *Robustness to lighting variations*. Because of the attenuation and the disturbances in illumination intensity, the spectral intensity of the images may vary. Therefore, the absolute intensities are not reliable features. Instead, cross-channel spectral features, such as the differences and the histogram overlapping, tend to be more reliable.
- *Computation load*. The features that are computationally expensive are not suitable for real-time applications.

In addition to the above factors, other issues such as the invariability to scale, transform, and rotation, also need to be considered in feature selection.

2.3.2.4 PNN Classifier

Various classification tools can be employed. Among these, probabilistic neural networks (PNN) are effective for supervised pattern recognition ([38]). The PNN with Gaussian kernels is adopted. It has the 4-layer feed-forward structure as illustrated in Figure 10. The input layer accepts length-p input feature vectors, where p is the dimensionality of the feature space. The nodes in the second layer are divided into K groups, where K is the total number of classes. The kernel in the i^{th} node in the j^{th} group is defined as a Gaussian basis function centered at $\underline{X}_{ji}$ with the covariance matrix Σ, where $\underline{X}_{ji}$ is the training pattern and Σ is the matrix of smoothing factors. When Σ is a diagonal matrix with equal diagonal elements, the Gaussian kernel can be replaced with a radial basis function of the

Euclidean distance between the input vector and $\underline{X}_{ji}$. The third layer has *K* nodes and each node estimates a class conditional *PDF* using a mixture (weighted sum) of Gaussian kernels. The fourth layer makes the decision based on the Bayes decision rule.

PNN is adopted in the proposed general methodology because it has the following advantages over other types of neural networks.

- The structure is simple and easy to understand.
- The training process is fast.
- The decision boundaries can be conveniently adjusted in order to make the trade-off between the *DR* and *FAR* so that the system requirements are met.
- The confidence level can be computed from the probability output for each input feature vector. In case the confidence level is low, segmentation can be repeated when the recursive segmentation and classification scheme is adopted.
- Look up tables (LUT's) can be used to further increase the speed.

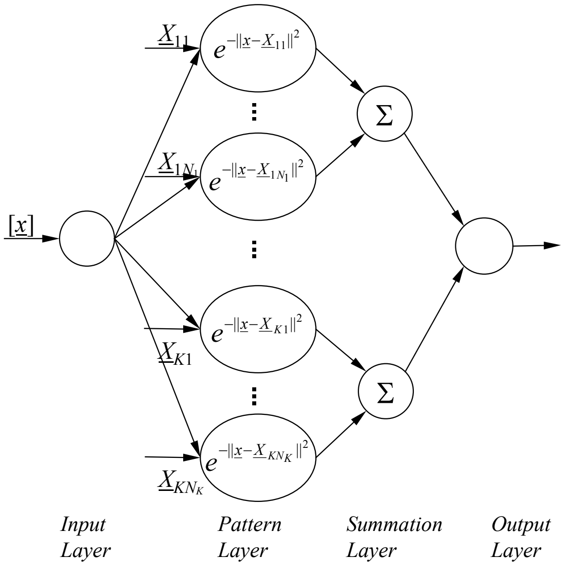

$\underline{X}_{11}$
$e^{-\|\underline{x}-\underline{X}_{11}\|^2}$
$\underline{X}_{1N_1}$
$e^{-\|\underline{x}-\underline{X}_{1N_1}\|^2}$
$[\underline{x}]$
$e^{-\|\underline{x}-\underline{X}_{K1}\|^2}$
$\underline{X}_{K1}$
$e^{-\|\underline{x}-\underline{X}_{KN_K}\|^2}$
$\underline{X}_{KN_K}$
Σ
Σ
Input Layer
Pattern Layer
Summation Layer
Output Layer

Figure 10. The structure of the probabilistic neural network with *K* classes.

CHAPTER 3

AN APPLICATION — REAL-TIME FAN BONE INSPECTION OF DEBONED POULTRY PRODUCTS

3.1 Problem Description

The project of automated inspection of fan bones in deboned poultry meat serves as a test bed for the developed methodology. The objective of the project is to detect the deboned chicken meat that contains fan bones (fan-shaped surface bones) using a MV-based inspection system. The chicken parts move on the conveyor belt at 60 feet per minute and the inspection is real-time and fully automated.

High-quality images are acquired using the on-line imaging system with the diffusive uniform lighting. The image of the deboned chicken breast contains multiple features, such as meat, fat, blood, bruise, bone, white membrane, and so on. In visible band images, the contrast between fan bone and its surrounding tissue is normally prominent. However, there are other spots on the image that may appear similar in color and shading to bones. These include shadows and edge characteristics that must be distinguished from fan bones. The products exhibit considerable variability in a variety of characteristics, including size, color, thickness, and hardness of bones. Moreover, depending on its location and orientation, the fan bone may not be visible to the camera.

The image database consists of more than 3000 color images taken on a lab-scale facility and another 3000 images from β-test in several poultry plants. To test the inspection system, 280 sample images (137 fan bone-contaminated and 143 bone-free) acquired in the lab under the same imaging conditions were used. Since the samples were gathered over a period of time from different plants, both the samples and the defects illustrated considerable variances.

3.2 Vision-Based Inspection Algorithm

Deploying the inspection methodology and following the design guidelines in last chapter, a vision-based inspection software algorithm is developed based on color imaging. The algorithm is shown in Figure 11 with more details presented in [13].

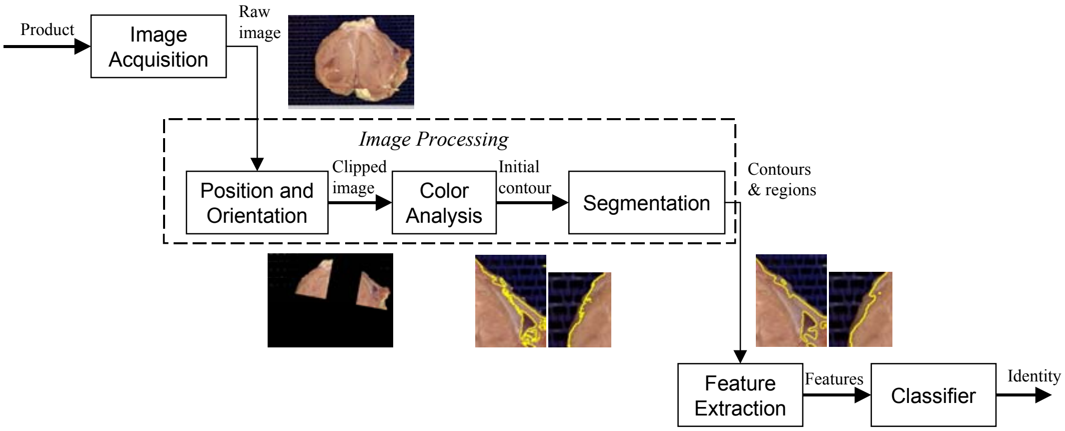


Figure 11. Visual-based inspection of fan bones.

Two modules are developed for preprocessing: position and orientation location and color analysis. The former removes the background (conveyor belt) and isolates the regions where fan bones might be according to the position and orientation of the sample in the image. The later selects the suspect pixels from the isolated regions based on the spectral characteristics of each pixel ([44]). Then the region-based snake algorithm (0) is adapted to segment the neighborhood regions of the suspect pixels. The snake algorithm uses region-based curve evolution equations to "pull apart" the mean intensity values between the interior of the contour and the background. A penalty term on the total arc length of the contour prevents the contour from wrapping around noise. The level set method in [33] is adopted in implementing the snakes in order to handle the topological changes of contours such as merging and splitting.

After 50 iterations of evolution, the contour evolves to capture three types of darker regions (Figure 12 (a)): *fan bone*, *edge*, and *shadow*. *Fan bone* regions are the regions that partially or completely catch fan bones (Figure 12 (b)); *edge* regions refer to the transition regions from the meat to the background (Figure 12 (c)); and *shadow* regions include dark meat, topological variations on the meat surface, bruises, etc. (Figure 12 (d)). Both *edge* and *shadow* regions are *non-fan* regions and need to be distinguished from *fan bone* regions by the classifier.

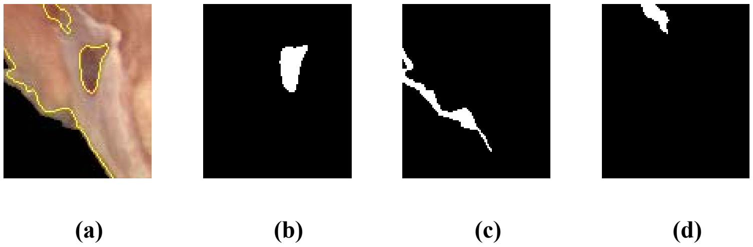


Figure 12. The result of the snake routine on a clip of chicken part image. (a) Final contour (b) A *fan bone* region (c) An *edge* region (d) A *shadow* region.

Various color and shape features have been tested to select the features with good distinguishibility and high robustness to color variations and segmentation errors. The selected features include the distance from the region to the background (conveyor belt), mean intensity differences of any two of the red, green, and blue channels, histogram overlapping between red and blue channels, circularity factor (the ratio of squared perimeter to area), and aspect ratio. Finally, the PNN classifier is adopted for the classification purpose.

The snake algorithm turns out to be efficient and accurate when the fan bone has good contrast with respect to its neighboring tissues. However, it fails when the contrast is not pronounced. Figure 13 is a typical example of segmentation failure, where under-segmentation occurs around the fan bone region. The failure is caused by the penalty term on the total arc length and the closeness of fan bone to the transition region. The under-segmentation totally changes the shape and spectral signatures of the region and thus causes misclassifications.

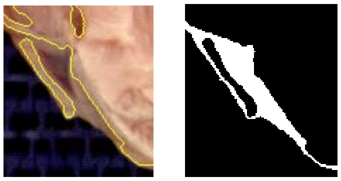


Figure 13. **An unsuccessful example of the snake routine.**

To show how this inaccuracy affects the overall performance, we visually assessed each *fan bone* region and classified it as *well-segmented* or *poorly-segmented* based on the closeness of the final contour to the true boundary of the fan bone. Of the 204 *fan bone* regions, 154 are segmented accurately, thus the segmentation accuracy is 75.5%. Figure 14 plots the normalized histograms of one feature value: the distance from the mass center of a region to the background, for *edge* regions, *well-segmented fan bone* regions, and *poorly-segmented fan bone* regions, respectively. The *poorly-segmented fan bone* regions have more overlapping with the *edge* regions than the *well-segmented* do, which results in a low detection rate of 40.9% on the *poorly-segmented fan bone* regions as opposed to 96.3% on the *well-segmented* ones (Table 2).

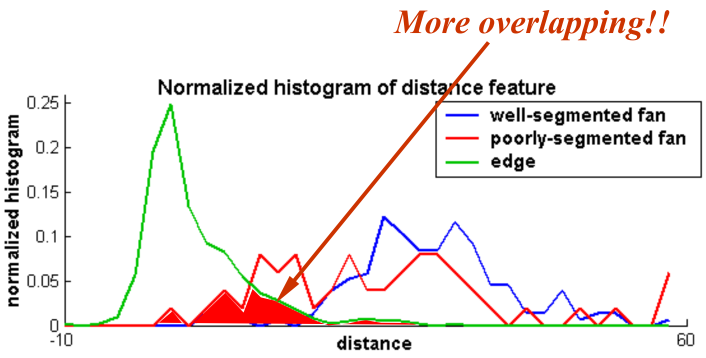

More overlapping!!

Figure 14. The normalized histogram of the distance feature.

Table 2. The classification results of *fan bone* regions.

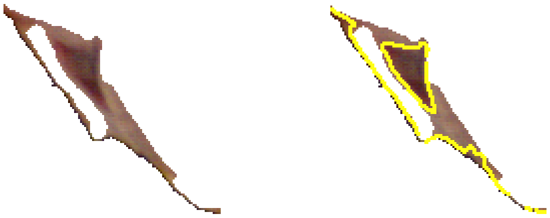
Region type	Well-Segmented	Poorly-Segmented
Total	82	22
Detected	79	9
Detection rate	96.34%	40.91%

To improve the performance on the sample images where fan bones cannot be segmented successfully, the recursive segmentation and classification scheme is applied. The confidence level (L_k) that a region belongs to class k is computed from the probability output of the PNN as

$$L_k = \min_{j \neq k}(P_k - P_j), k, j = 1,2,\cdots,N \tag{5}$$

where N, the total number of classes, equals 3 in this problem, and P_k is the probability that a region belongs to class k.

A maximum of two iterations of segmentation and classification is performed. The classifier for the first iteration is the PNN while the second is a set of heuristic rules. After the 1st iteration, the confidence level output of the PNN is checked. If it is lower than the threshold of 5%, segmentation is repeated by the same active contour algorithm using different parameters, initial contours, and ROI. Figure 15 is the result of reapplying the snake algorithm to the under-segmented region in Figure 13. By taking the under-segmented clip as a bimodal image, the snake algorithm successfully separated the fan bone from the transition region.



(a) raw image clip **(b) final contour**

Figure 15. The result of reapplying the snake algorithm to the under-segmented region in Figure 13.

3.3 Lab-Scale Test Results

Totally 834 regions (204 *fan bone*, 402 *edge*, 228 *shadow*) are generated by the snake algorithm from the 280 chicken images. Numerical features are extracted from each region. 100 samples from each class are selected arbitrarily to train the PNN classifier while the others are left for validation. The classification result without using recursive segmentation and classification is shown in Table 3. The overall accuracy achieved is 90.4%.

Table 3. **The classification results without using the recursive segmentation and classification scheme.**

***Fan bone* regions**		***Non-fan* regions**	
Total	**Correct**	**Total**	**Correct**
104	88	430	395
Detection rate = 84.62%		False alarm rate = 8.14%	

Applying the two-iteration recursive segmentation and classification scheme, the classification results after the first and the second stages for all regions and *fan bone* regions only are shown in Tables 4 and 5, respectively. Compared with the results in Table 3, the detection rate is increased by 7.7 points while the false alarm rate is decreased slightly. A significant increase from 40.9% to 77.3% is observed in the detection rate on the *poorly-segmented fan bone* regions after using the recursive scheme. The overall accuracy is increased from 90.4% to 92.88%.

The algorithm is implemented on a personal computer (Pentium III, 933MHz). The computation time for the whole process is about 800 milliseconds.

The experiment results demonstrated that the proposed methodology is effective in detecting abnormalities with certain spectral and shape features at a fast inspection speed.

Table 4. The classification results using the recursive scheme.

	***Fan bone* regions**				***Non-fan* regions**			
	Total	**Correct**	**Incorrect**	**Unclassified**	**Total**	**Correct**	**Incorrect**	**Unclassified**
1st iteration	104	84	5	15	430	367	15	48
2nd iteration	15	12	3	/	48	33	15	/
	Detection rate = (84+12)/104 = 92.3%				**False alarm rate** = (15+15)/430 = 6.98%			

Table 5. The classification results for *fan bone* regions using the recursive scheme.

	Well-segmented				***Poorly-segmented***			
	Total	**Correct**	**Incorrect**	**Unclassified**	**Total**	**Correct**	**Incorrect**	**Unclassified**
1st iteration	82	78	0	4	22	6	5	11
2nd iteration	4	1	3	/	11	11	0	/
	Detection rate = (78+1)/82 = 96.34%				**Detection rate** = (6+11)/22 = 77.27%			

CHAPTER 4

MULTISENSORY IMAGE SEGMENTATION USING ACTIVE CONTOURS (SNAKES)

4.1 Motivation of Multisensory Image Segmentation

It has been shown in Section 3.2 that the segmentation accuracy has a significant effect on feature distributions and classification performance in object recognition applications. However, segmentation on monosensory images may not produce satisfactory performance due to the intrinsic ambiguity and incompleteness associated with the data. In monosensory image segmentation, both over- and under-segmentation can happen in addition to false regions and edges. Various techniques have been developed to increase the segmentation accuracy on monosensory images by combining segmentation results generated from different algorithms, as have been reviewed in Section 2.1.2.2. These techniques are effective in overcoming the limitations of individual algorithms. However, when the local contrast of the object with respect to the background is not good, it is highly possible that all algorithms will fail. Besides, by deploying these techniques, the computational cost increases significantly and thus, they cannot be used in real-time applications. Images acquired by different sensors are generally partially redundant and partially complementary, which can be used to reduce the imprecision and to interpret the scene more accurately. Unfortunately, the multisensory image segmentation problem is not a straightforward extension of its monosensory counterpart to multiple images. The algorithm must make smart decisions on how to eliminate the redundancy, to include the complementary information, and to resolve the possible conflict.

Multisensory image segmentation refers to the process of partitioning images from different sensors. By combining different modalities in segmentation, we expect to achieve the following goals:

- Locate more accurate boundaries for the objects in the scene.
- Eliminate false edges/regions as many as possible.

Note that for the false edges/regions that cannot be totally eliminated, the significant difference between the segmentations on individual modalities can be utilized by the pattern recognition algorithms to separate false edges/regions from true edges/regions.

One way of doing multisensory image segmentation is to employ a certain image fusion technique to merge all images into one, on which the segmentation is then performed. This way, the image merger handles the information of various natures. However, the merging operation may increase the noise power and generate artifacts, which makes the segmentation more difficult. Besides, the merged image may be even harder to interpret. Finally, many merging algorithms involve feature detection such as segmentation and edge detection. In contrast to this approach, the multisensory segmentation has several advantages. First, the amount of data to be processed is greatly reduced by focusing on the higher level representation of images. Second, it is not necessary to pay special attention to integrating information for visualization, which simplifies the task significantly.

In the snake-based curve evolution methods, the contour evolves according to the optimal flow derived from the energy functional. Segmentation errors occur when the images do not satisfy the assumptions based on which the energy functional is constructed. Because this type of method is local in nature, the curves tend to get trapped by unexpected features before they reach the true edges, causing either under- or over-segmentation errors. Since the unexpected feature that entrapped the contours may show itself differently in images acquired by other sensors, it may function as a driving force to push the contour toward the true edge and thus reduce the segmentation errors. In the proposed model, the snake is driven by the combined force that is the weighted sum of several forces, each derived from the energy functionals that are specially designed for individual images. This method is applicable to cases where the objects in the scene exhibit totally different views in images. *A priori* information of the scene and sensors can be employed to further speed up this process so that the algorithm is more suitable for real-time applications.

In this chapter, the general concepts of multisensory images and image fusion are firstly introduced, followed by a brief review of multisensory image segmentation. Then, a real-time multisensory image segmentation approach is presented with results on both synthesized and real images. Finally, the approach is applied to segmentation of the color and x-ray images of bone-contaminated poultry meat, and the results are presented.

4.2 Background

4.2.1 Multisensory Images

Multisensory images are a type of vector-valued images. An M-vector-valued image is defined as $I(x, y)$: $R^2 \rightarrow R^M$, with components $I_i(x, y)$: $R^2 \rightarrow R$, $i = 1, 2, \ldots, M$. The value of the image at a given point (x, y) is a vector in R^N. The most commonly seen vector-valued images are color images. Others include those obtained in different image sensors (modalities), as in medical and LANDSAT applications. Vector-valued images can also be generated from popular image processing operations, such as scaling, down/up sampling, and multiscale decomposition. Multisensory images specifically refer to images obtained from different image sensors. A significant difference from other types of vector images, such as the images resulting from multiscale decomposition, is that multisensory images often contain information conflict, either because of different sensor characteristics or registration error. Therefore, the segmentation algorithm must have the capability of resolving contradictory information.

Three multisensory image examples are given in the following figures. The first example is shown in Figure 16, where a pair of registered CT and MRI images of brain is shown. The second is the infrared and visible road images shown in Figure 17. The third is the x-ray and color images of deboned poultry meat in Figure 18.

The images in each example show different image features about the scene that is being imaged. The differences manifest themselves as variances in intensity, noise, and texture. To simplify the problem, we do not consider the texture features in this research. However, images with texture information can be transformed using Fourier Transform or Wavelet Transform to obtain multiple channels, which can then be processed using the same segmentation method.

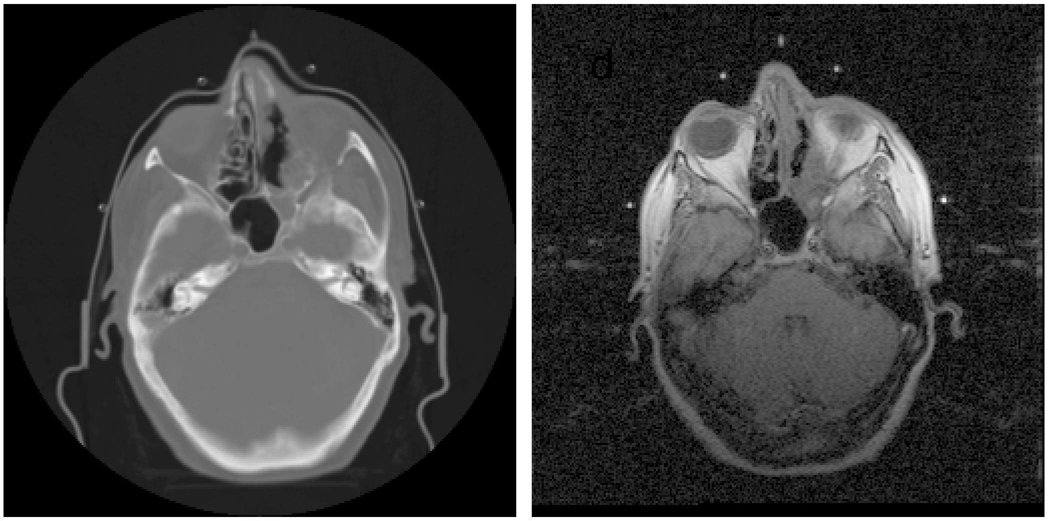


Figure 16. CT (left) and MRI (right) images of brain.

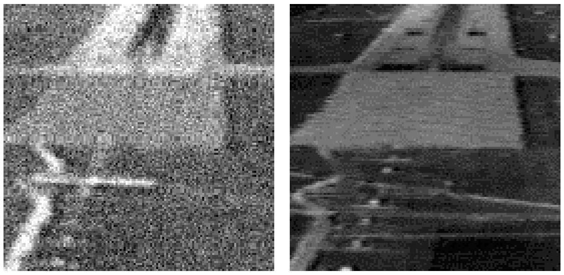


Figure 17. Infrared (left) and visible (right) road images.

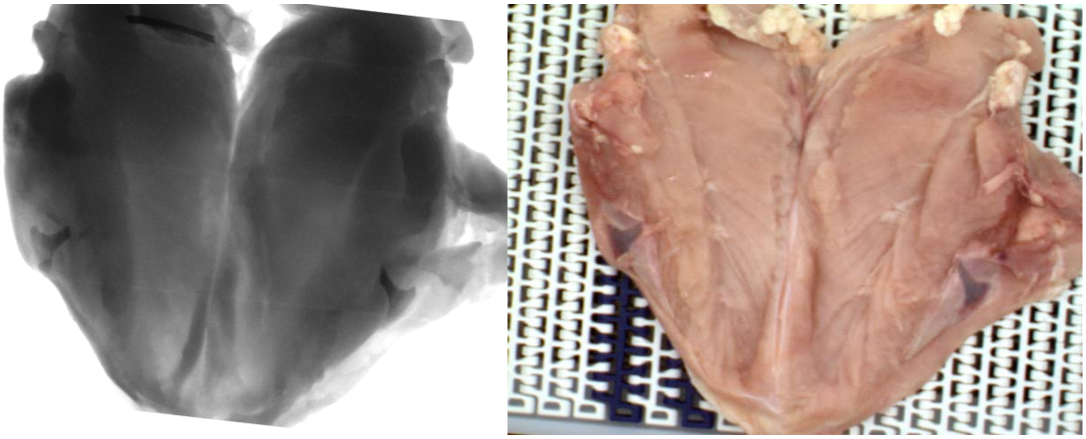


Figure 18. X-ray (left) and color (right) images of a deboned poultry meat sample.

4.2.2 Fusion of Multisensory Images

Although a large amount of literature use the term image fusion to refer to pixel-level (low-level) image merging, in this research we stick to a more general definition of image fusion as the assimilation of information acquired by two or more sensors viewing the same scene ([32]). Fused results can be images, which are not necessarily the visual pictures, or symbols that describe the scene. The fused results may only be understood on the basis of some specific knowledge of the sensor data ([26]).

The multisensory images have the following properties: imperfection, complementarity, redundancy, and heterogeneity. Image fusion needs to (a) increase completeness by including complementary information, (b) reduce uncertainty and imprecision, (c) eliminate redundancy, (d) resolve conflicting information, and (e) obtain a precise and meaningful interpretation of the scene.

Normally, the image fusion problem is the four-step procedure shown in Figure 19: preprocessing, generation of image description, fusion, and generation of world description.

Depending on the level of description at which the information is fused, image fusion can be carried out at one or more of the following levels: pixel, feature, and decision.

- Pixel level: combine two or more spatially registered images into one enhanced image in which both the contrast and the signal-to-noise ratio are increased. In the combined image, features in individual source images are preserved or enhanced with minimal artifacts. Source images can be described in either spatial or frequency domain, and multiresolution techniques are widely used at this level.

- Feature level: source images are independently processed using intermediate level image processing algorithms. The obtained segmentation features, such as edges and regions, and numerical features and measurements, such as values of shapes, fuzzy measures and probabilities etc., are then merged using the mathematical tools provided by the framework in which the features are represented.
- Decision level: each source image is processed, classified, and then combined in a supervised or unsupervised manner.

Figure 20 depicts the data flow of fusion at different levels. Multisensory image segmentation is a feature-level (or intermediate-level) fusion technique. The fused result is a feature map indicating the positions of multisensory image features such as regions and edges.

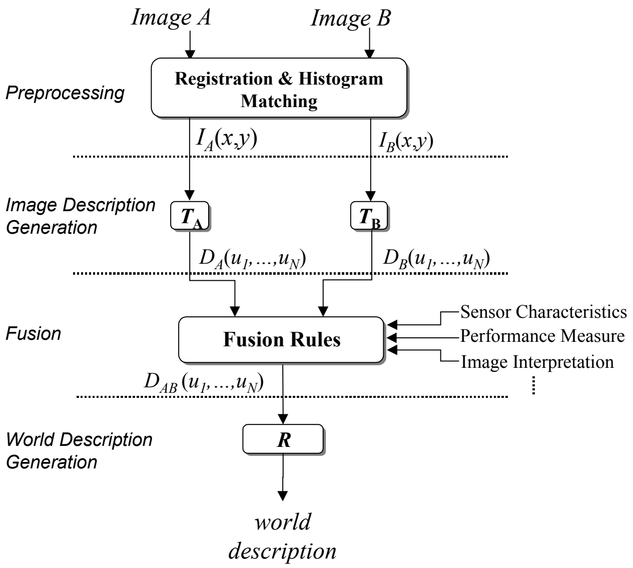


Figure 19. General procedure of image fusion.

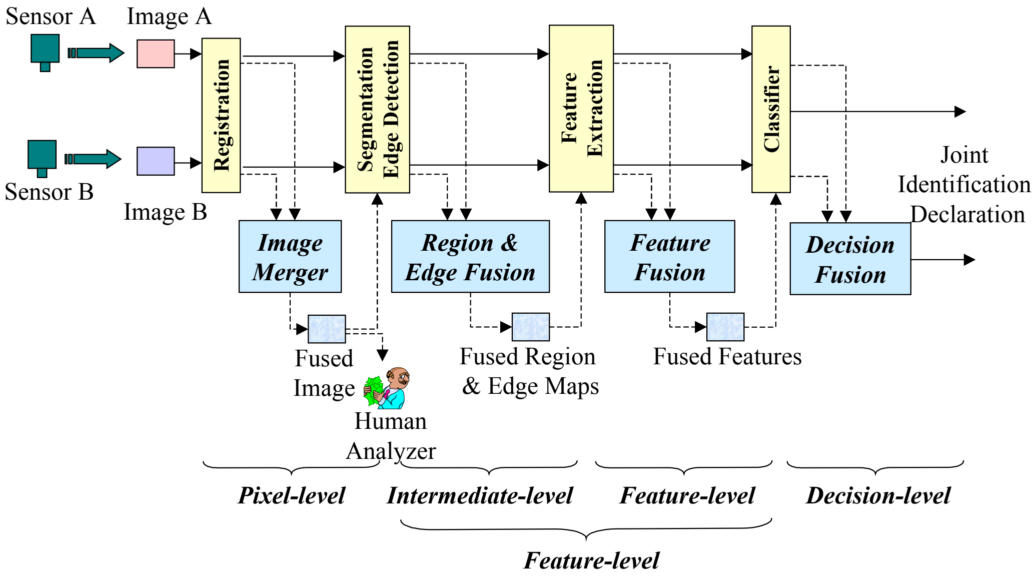


Figure 20. Different levels of image fusion.

4.2.3 Multisensory Image Features

Features of an image include regions, edges, and textures. Depending on the characteristics of physical sensors and target objects, the definition for edge and region on multisensory images may vary. Bonnin [1] defines a multi-spectral edge point as a pixel where there are important variations in at least one direction in the neighborhood of a property, in at least one spectral image. This definition implies a logical *OR* between different spectral images of edge points. The displacement of an edge between two spectral images has to be taken into account and solved in the edge thinning and chaining steps. In the mean time, the false edges caused by artifacts and noises accumulate by applying logical *OR* directly. The majority voting rule in [32] eliminates false edges if they are observed only in a limited number of channels, while at the cost of missing possible important edge features. Therefore, fusion of edge maps is not a trivial problem.

Similarly, the region of homogeneity implies a logical *AND* between the homogeneity predicates in all spectral images. The homogeneity criteria are usually some thresholding of the homogeneity measures. The measures can vary largely with different sensors. Even for the same sensor image, the thresholds may change with illumination and the scene being imaged. The choice of the homogeneity measure and the adjustment of its thresholds are demanding tasks and are usually guided by the knowledge of the sensors' physical characteristics.

In order to get a more thorough understanding of the multisensory features, we need to compare carefully the behaviors of different sensors. Observing the multisensory image examples provided in 4.2.1, we find that the features in one image fall in one of the following categories with regard to the corresponding features in other images.

(a) *Redundant* or *compatible* features. The feature (edge/region) can be observed in all images, but the contrasts are different because of different intensity levels and noise powers. This can be observed from all three examples. For this type of features, in spite of different intensity distributions, edge positions and region boundaries across images are consistent. Therefore, the corresponding multisensory features are the same as the monosensory ones.

(b) *Complementary* features. The feature can be observed in only some modalities, such as the eyes in the MRI image in the brain example, the pulley bone in the x-ray image, and the fat in the visible band image of the poultry example. In this case, the multisensory features are obtained by *OR*-ing all monosensory edge features, or *AND*-ing all region features.

(c) *Contradictory* features. Obvious displacement can be observed among corresponding image features across images. This displacement is caused either by distinctive sensor characteristics or by registration error (misalignment) between images. To illustrate the registration error of fan bones in the poultry meat example and the displacement in the brain example, the original images are clipped and shown as the pseudo-color images displayed in Figure 21 (a) and (b), respectively. Note that this feature displacement is normally within the range of a few pixels. It is difficult to define multisensory features for contradictory monosensory features. Considering the ground truth for all monosensory features is the same (the scene being shot,) the corresponding multisensory features should be unique. From the image processing point of view, the multisensory feature in this case can be deemed as the most pronounced monosensory feature in all modalities. In practice, though, other definitions may be found more reasonable with prior knowledge on sensors and scenes.

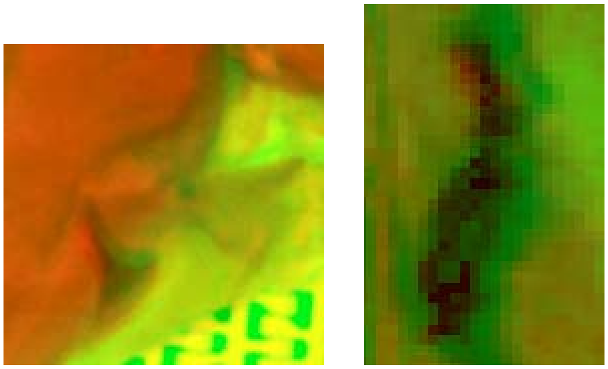


Figure 21. Example of the contradictory monochannel features.
Left: registration error in the poultry meat example; Right: displacement caused by sensor characteristics in the brain example.

With the above categorization, the task of multisensory image segmentation then becomes identifying the nature of monosensory features and locating their multisensory counterparts.

4.2.4 Overview of Multisensory Image Segmentation Methods

One common scheme for multisensory image segmentation, as shown in Figure 22, is to obtain features from individual sensor images, then combine these monosensory features using simple fusion rules or rules based on the mathematical frameworks of Bayesian Theorem, Fuzzy Theory, and Dempster-Shafer (DS) Theory. In order to utilize the rules in such frameworks, the fusion problem first has to be modeled in that framework. One application is to fuse SAR and optical images to achieve better performance in detecting urban areas ([16])[15]. The fusion of individually segmented images is modeled as a nonlinear optimization problem, where the objective function is the sum of two terms: the mismatch between the two fused segmentation maps and the discrepancies between corresponding fused and original segmentations. The handling of unmatched monosource features are controlled by the penalization parameters on the second term. One concern about this method is that *ad hoc* calibrations of several parameters are necessary. Another issue is that the nonlinear optimization problem is hard to solve and can be too slow to be used real-time.

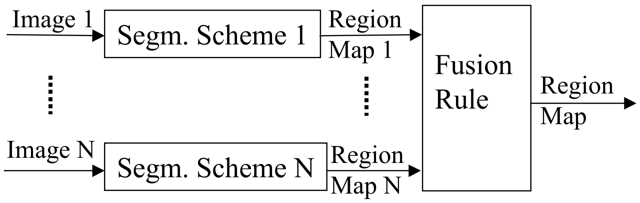


Figure 22. Fusion of individual segmentation results.

One can also build a segmentation map directly on all images, as shown in Figure 23. For example, in [21], multisensory data are classified pixel by pixel using DS theory to produce a labeled image (segmentation map). However, the segmentation map needs to be further refined to remove speckle errors. A work unifying snakes, region growing and Bayes/MDL for multiband image segmentation is derived from minimizing a generalized Bayes/MDL criterion using the variational methods ([49][48]).

Several active contour algorithms are proposed for segmentation of vector-valued images based on the partial differential equation (PDE) models. They are the natural extensions of their monosensory counterparts to vector-valued images. The extension of the Mumford-Shah functional to segmentation of color images is to be found in [34]. A region grow algorithm based on the simplified Mumford-Shah functional is developed in [25] for segmentation of the multichannel images generated by textured images. Even if a region grow scheme is adopted to increase the processing speed, this model is still too slow and cannot be used in real-time applications.

Extending the geodesic active contour model for single-valued images introduced in [4], the color geodesic snakes model is developed in [31] for vector-valued images, where the notion of vector edges are defined based on the classical results on Riemannian geometry. These vector edges are used to define the stopping edge function. The geodesic curve is then a minimal “color weighted” path. The behavior of the color snake in the presence of contradictory features needs to be investigated. Besides, the geodesic active contour model itself is sensitive to noise.

The scalar Chan-Vese model is extended to vector-valued images in [7] by exerting a weighted-sum of the fitting error over each component of the vector-valued image.

Before applying this model, the weights need to be chosen, which requires the a priori knowledge about the image such as occlusion and noise.

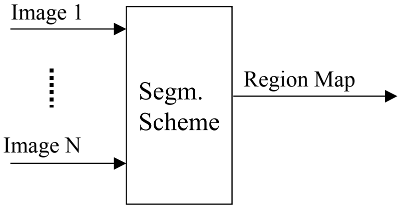


Figure 23. Multichannel image segmentation.

In the following section, we propose a general and flexible framework for snake-based segmentation of vector images. It combines multiple forces into one at each contour point. The combined force then drives the evolution of the snakes to produce a segmentation of the vector image. The developed scheme is fast and efficient in resolving conflict. Thus, it is suitable for real-time applications which involve multisensory image segmentation task. Although this algorithm is designed for the purpose of multisensory image segmentation, it can readily be applied to the segmentation of other types of vector images.

4.3 An Active Contour-Based Approach for Multisensory Image Segmentation

4.3.1 Model Description

In this section, we propose a novel algorithm for multisensory image segmentation based on the popular and powerful segmentation method of *snakes*. The basic idea is illustrated in Figure 24, where there is only one set of active contours $\vec{C}$ defined for all image attributes. For each attribute, distinctive energy functional can be constructed. Each point on the contour evolves according to the force combined from all components, each being the force that optimizes its corresponding energy functional. When combining forces, the weight on each image is determined based on local contrast and noise in that image. The following is a detailed description of the algorithm.

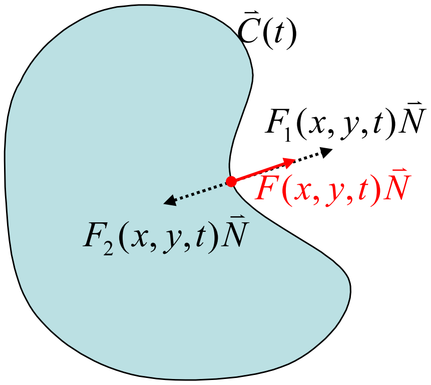


Figure 24. The snakes for multisensory image segmentation.

An appropriate energy functional, E_i, can be designed for each image attribute $I_i(x, y)$, according to the characteristics of sensor i, $i = 1, 2, \ldots, M$, where M is the total number of images. The energy functionals can be different, but each determines an optimal force $F_i(x, y, t)$ at time t and position (x, y) on the contours $\vec{C}$ along its normal direction $\vec{N}$:

$$\left.\frac{\partial \vec{C}(x,y,t)}{\partial t}\right|_{I_i} = [F_i(x,y,t)\vec{N}]dt\,,\ i = 1, 2, \ldots, M \tag{6}$$

where dt is the step size. Then the individual forces are combined through weighted averaging as follows in order to obtain a unique force at time t and position (x, y):

$$F(x,y,t) = \sum_{i=1}^{M} w_i(x,y,t)F_i(x,y,t),\ \sum_{i=1}^{M} w_i(x,y,t) = 1 \tag{7}$$

where w_i is the weight on the i^{th} image.

The weight selection is critical since it determines how different types of features are handled. Note that although $F(x, y, t)$ is expressed as the weighted sum of all forces, depending on the weight selection, the equation can have different interpretations. For example, if the weight on the strongest force is set to 1 but all others set to 0, then the weighted averaging of all forces becomes selecting the strongest component:

$$F(x,y,t) = F_i(x,y,t), \text{ where } |F_i(x,y,t)| = \max_{j=1,\cdots,M}(|F_j(x,y,t)|) . \tag{8}$$

However, since the forces generated by different energy functionals may not be comparable in amplitude, it may not make sense to compare forces directly.

Weights can vary as the contours evolve. When they are set constant, the combined force equals the one derived from the energy functional that weighted averages all functionals using the same weights.

A reasonable way in assigning the weights is to compare the local feature information in individual channels. Generally, the following guidelines need to be followed.

- For redundant monosensory features, equal weights are used.
- For complementary monosensory features, the sources where the features are observed are assigned higher weights.
- For contradictory monosensory features, the sources with more reliable information are given higher weights.

The key issue then becomes identifying the nature of the monosensory features and resolving information conflicts according to the local information in each source. When we have little prior knowledge of the sensor characteristics and the scene, the weight computation becomes a demanding task. In the following sections, a systematic method is proposed in obtaining weights according to a metric which measures both contrast and noise. This metric is based on the statistical model of noise-corrupted bimodal grayscale images

4.3.2 Gaussian Mixture Model of Noise-Corrupted Bimodal Images

The simplest non-trivial scene contains an object against a background. Without considering texture features, the noise-free image $I_0(x, y)$ will only have two intensity levels: u and v, with areas A_u and A_v, respectively. Define p_u and p_v as the probabilities of the foreground and the background:

$$p_u = \frac{A_u}{A_u + A_v}, \text{and } p_v = 1 - p_u, \tag{9}$$

then $I_0(x, y)$ has the following binary distribution

$$p(I_0) = p_u \delta(I_0 - u) + p_v \delta(I_0 - v). \tag{10}$$

Assuming the noise $n(x, y)$ generated by the sensor and the circuits is additive Gaussian with mean zero and variance $\sigma^2(x, y)$, the noise-corrupted bimodal image $I_n(x, y)$ then becomes

$$I_n(x, y) = I_0(x, y) + n(x, y), \tag{11}$$

where the noise $n(x, y)$ has the following probability density function (PDF)

$$p(n) = N(0, \sigma(x, y)) = \frac{1}{\sqrt{2\pi}\sigma(x, y)} e^{-\frac{n^2}{2\sigma(x,y)^2}}. \tag{12}$$

Note that the noise power σ^2 at each point (x, y) can be either dependent or independent of the intensity $I_0(x, y)$, depending on where the noises are generated. Assuming that the noise power is constant across the image, the PDF of the intensities in a noise-corrupted bimodal image is then

$$p(I_n) = p_u N(u,\sigma) + p_v N(v,\sigma), \qquad \textbf{(13)}$$

and the mean and variance of this distribution are $p_u u+p_v v$ and $p_u p_v(u-v)^2+\sigma^2$, respectively.

4.3.3 A New Metric of Local Contrast and Noise

When combining multiple forces, the weight on each force component needs to be determined. Since segmentation performance is greatly affected by the contrast $\frac{|u-v|}{v}$ ([36]), and the noise power σ^2, heavier weight needs to be assigned to the image attribute with higher contrast and lower noise. Therefore, a measure that quantifies both contrast and noise is desired.

In the literature of image merging, or pixel-level image fusion, the term "activity level" has been widely adopted in determining which attribute is to be given higher weight when merging multiple images. A common method is to calculate the activity level of image I_i at each point (x, y) based on (i) the transform coefficient at that point, (ii) a group of coefficients within a window (3x3, or 5x5), or (iii) the coefficients within a region (a window with an odd shape) centered around that point. The coefficients are obtained from a certain type of multiscale decomposition (MSD) [48]. The point-by-point measures are very sensitive to noise, while the region-based measures involve feature detection during the computation. The window-based measures are claimed to be less sensitive to noise than the point-by-point ones, but still higher noise power generates larger activity values. Take the window-based measures called *match* and *saliency* measurements proposed by Burt ([3]) as an example. The match metric determines the mode of combination (*selection* or *averaging*) at each sample position: the most salient component patterns are selected where the source images are distinctly different, while sources are averaged where they are similar. The salience metric determines which source pattern is chosen in the selection mode. The amplitude of the coefficients of the Laplacian pyramids in a point's neighboring window is used in computing both measures at that position: the weighted square sum being the salience, while a normalized correlation being the match metric. The weights are then computed from these two metrics, with the larger weight always assigned to the source with larger salience metric. Figure 25 illustrates the weights computed using Burt’s method. Obviously, the weights generated by the noisier image (bottom) are much larger than those by the less noisier one. Therefore, the activity level is not a suitable measurement for our purpose and a novel one must be defined such that it increases with contrast and decreases with noise power.

An ideal choice of such measures is the *Fischer's Discriminant Ratio* (*FDR*) between the intensity distributions inside the foreground and the background regions, since the image segmentation problem is essentially a classification problem and the *FDR* indicates how well the two modes can be separated. The *FDR* between the two modes, denoted by R_F, is defined as

$$R_F = \frac{(u-v)^2}{2\sigma^2}. \tag{14}$$

To compute R_F, the means and variances of the two modes have to be estimated from the distribution of the image intensities. Optimal estimation methods for Gaussian Mixture parameter estimation can be found in [40]. In this research, instead of estimating mean and variance of each Gaussian mode, we propose a novel measure that can be computed directly from the intensity distribution. It will be shown that this measure, denoted by M_{FDR}, increases with R_F and thus indicates how well the image can be segmented.

Combining Shannon's entropy $H(I_n)$ and the overall variance Var of an image $I_n(x, y)$ in the following way,

$$M_{FDR}(I_n) = \log(Var) - 2H(I_n)\,, \tag{15}$$

we can show, with some derivation, that

$$M_{FDR} = \log(p_u p_v R_F + 1) - \log(2\pi e) - 2\lambda(R_F)H(I_0)\,, \tag{16}$$

where

$$H(I_0) = -p_u \log p_u - p_v \log p_v \tag{17}$$

is the entropy of the noise-free bimodal image I_0, and $\lambda(\cdot)$ is a monotonically increasing function of R_F, approaching 0 as $R_F \to 0$ and 1 as $R_F \to \infty$, as shown in Figure 26.

Obviously the first term in M_{FDR} has the desired property of increasing with R_F. The last term, however, tends to decrease by an amount of no larger than the noise-free image's entropy. In Figure 27, M_{FDR} is plotted against R_F and p_u for the special case of $u = 0$ and $v = 1$. It can be seen that for a given p_u, M_{FDR} increases with R_F and is roughly constant when p_u is greater than 0.1. This can be seen more clearly in Figure 28. The curve does drop when p_u approaches 0.5 for small R_F's (<=5). However, the variance is tiny ($<10^{-2}$) noticed that M_{FDR} is plotted in the logarithm scale.

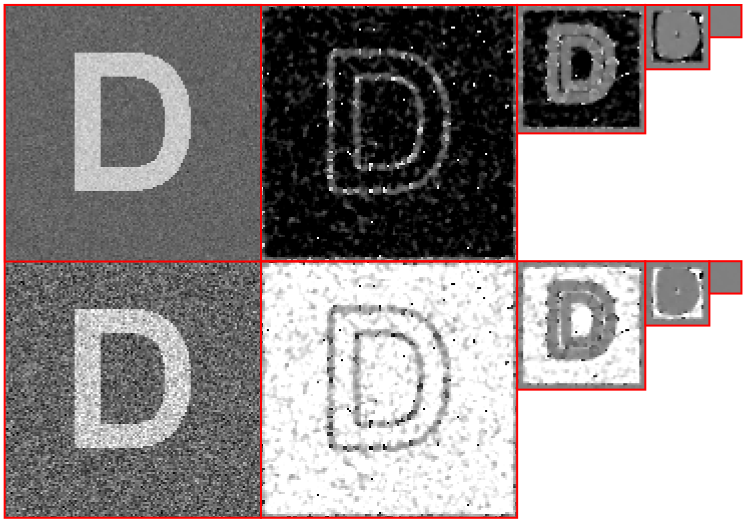


Figure 25. The weights computed from salience and match metrics.

Top: $\sigma = 10$; Bottom: $\sigma = 30$. From left to right: original image and weights computed at each level of the level-3 Laplacian pyramids.

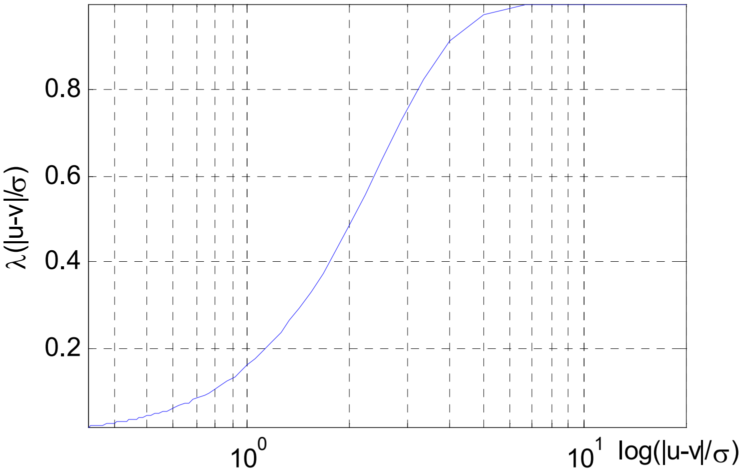


Figure 26. $\lambda(|u\text{-}v|/\sigma)$ vs. $|u\text{-}v|/\sigma$.

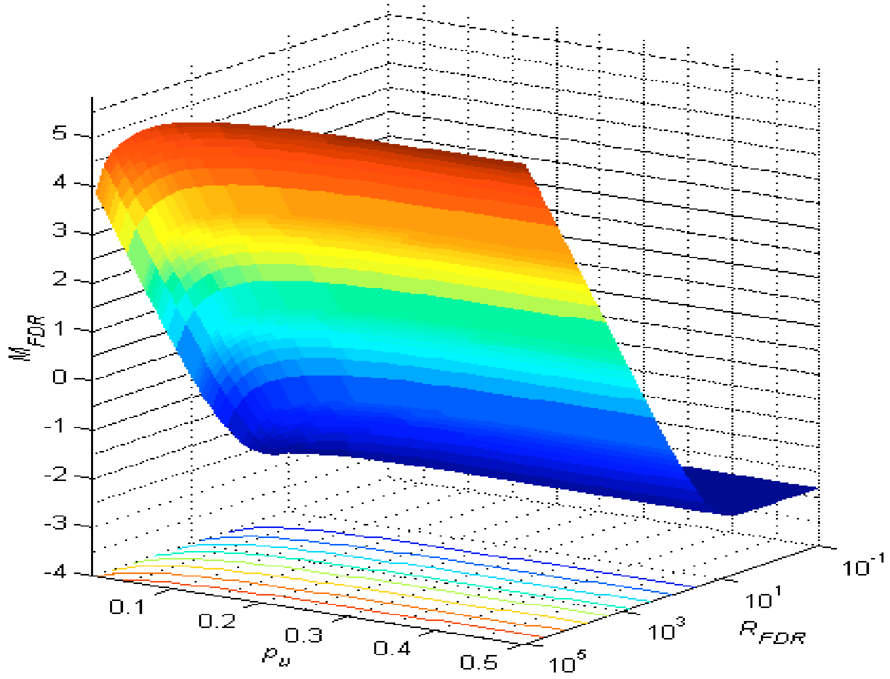


Figure 27. A 3D plot of M_{FDR} versus R_F and p_u (u=0, v=1).

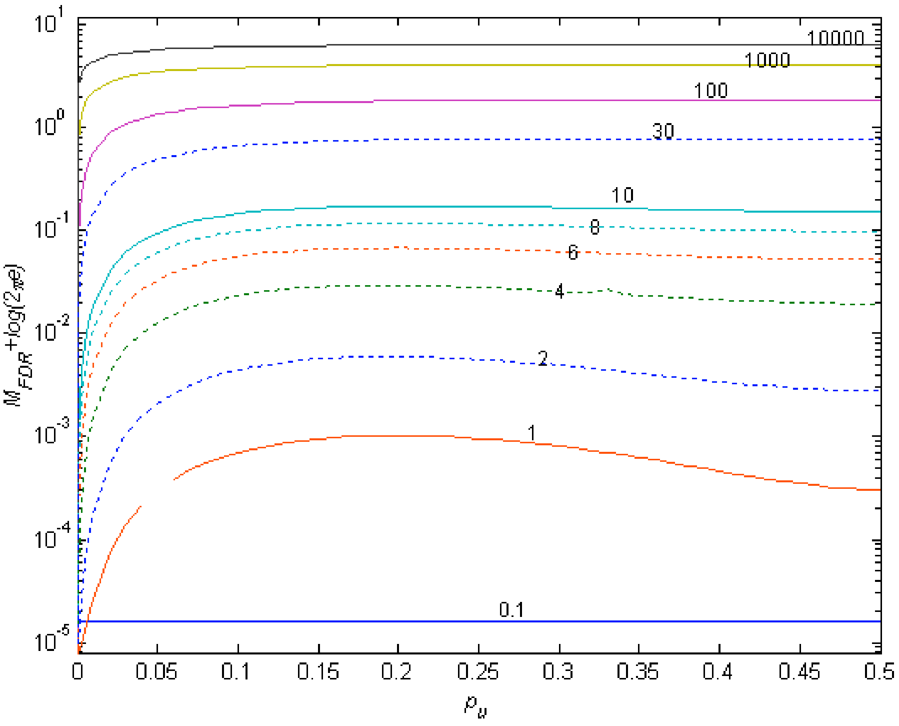


Figure 28. M_{FDR} versus p_u when u=0, v=1. The number on each curve is R_F.

It must be pointed out that M_{FDR} increases with p_u when p_u is small. As a result, a very small object with higher R_F would yield a greater M_{FDR} than a larger object with considerably lower R_F. In case this is found to be undesirable in applications, noise filtering techniques can be applied to increase R_F.

Note that the above derivations are all based on images with continuous values in the real domain. Since images are normally digitized into a fixed number of levels (e.g. 256 levels with 8-bit depth), the PDF must be replaced with probabilistic mass function (PMF). However, the number of bins N_{bins} being used will affect the value of M_{FDR}. Also a digitized image have saturation effects toward both ends, which causes deviation of the calculated M_{FDR} from the ideal M_{FDR}.

Common image operations such as filtering and histogram equalization will affect M_{FDR} too. Low-pass filtering of an image reduces an image's information and thus, increases M_{FDR}. Histogram equalization alone does not affect the entropy term, but the overall variance changes and M_{FDR} will change accordingly. Before weights are calculated, images from different sensors need to be preprocessed appropriately so that their M_{FDR}'s are matched (comparable to each other).

4.3.4 Weight Calculation

The weight w_i on the i^{th} image can be calculated based on the new feature information measure M_{FDR} defined above. The normalized linear ratio is the simplest method in calculating weights:

$$w_i = \frac{M_{FDR\,i}}{\sum_{j=1}^{N} M_{FDR\,j}} . \tag{18}$$

Another choice is, for two image attributes, to use the sigmoid functions:

$$\begin{cases} w_1 = \dfrac{1}{1+e^{-\beta(M_{FDR1}-M_{FDR2})}} \\ w_2 = 1-w_1 \end{cases} \quad \textbf{(19)}$$

where β controls the ramp of the sigmoid function.

In practice, images may contain multiple features and cannot be approximated using the bimodal image model developed in the last section. In this case, all images tend to contain prominent features and the M_{FDR}'s calculated on all images are most likely equal. Then the proposed scheme degrades to averaging forces all through the curve evolution process, which results in the missing of important local features, as can be seen in the upcoming sections of experiments. Therefore, in practice, we have to calculate weights at each pixel using only the information inside the neighboring window of that pixel. This is a more practical and effective approach considering that the window of a reasonable size contains only limited features and is very close to the bimodal model. The combined force at each contour point is then obtained using the weights computed at that point.

Note that there are some variations to this localized scheme. The first variation is that instead of varying weights at different locations along the contour, we can use the neighboring information of the entire contour to determine a fixed weight that is applicable to the whole set of contour points at a specific time. This has the benefit that the contour will move in a more consistent manner. Another variation is that, instead of weighted averaging all forces, select the force with the greatest M_{FDR}. To avoid switching frequently among different attributes, one can update the weights only after a certain number of iterations. This generates the scheme depicted in Figure 29. Since each segmentation starts from the previous segmentation, the convergence is faster than segmenting each modality independently. Besides, for conflicting features, *i.e.*, features with obvious displacement across channels, this scheme captures accurate boundaries in at least one channel and thus, avoids resolving the conflicts. This might produce more desired results in practical applications.

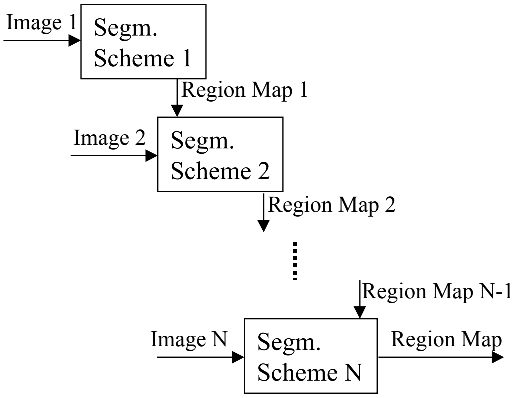


Figure 29. Segmentation using selection mode.

4.4 Experiments and Performance Assessment

The developed algorithm is tested on a set of test images before being applied to real medical and aerial images. In this section, the process of each experiment is described and the results are presented.

4.4.1 Experiment on Test Images

In order to compare the segmentation performance when using different weights, a seed binary image is created and various operations are applied on this image to emulate different sensor activities. The seed image I_0, as shown in Figure 30, is a bimodal grayscale image with a darker foreground (a capital letter D) against a background.

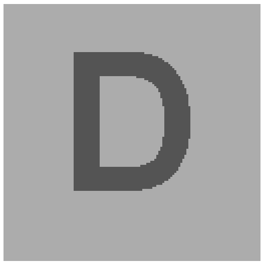


Figure 30. The seed gray-scale test image.

The seed image is duplicated to create a bichannel image, then the following operations are applied to each channel to emulate different sensor behaviors.

- Noise: add Gaussian white noise sequences with various noise powers to channels.
- Blurring: convolute the seed image with 2-D Gaussian kernels with various radii and variances. The blurring effect is the result of the point spread function (PSF) of the imaging system.
- Complementary features: move some of the features from one channel into the other.
- Registration error: shift the image by a certain number of pixels in one channel.
- Contradiction: change the feature in one channel by resizing and rotating it.

Using the above operations, a set of test images is generated, on which the following three weight computation methods are applied: constant equal weights (CEW), constant weights computed from M_{FDR} of the entire image, or the global constant weights (GCW), and window-based local weights (WLW).

The energy functionals adopted for both channels in all examples are the binary flow in [47]

$$E_i = -\frac{1}{2}(u_i - v_i)^2, \quad i = 1, 2 \qquad \textbf{(20)}$$

where u_i and v_i are the mean intensities inside and outside the contour in channel i, respectively. This energy functional yields the following optimal force at each point (x, y) of the curve in channel i

$$F_i(x,y) = (u_i - v_i)\left(\frac{I_i(x,y) - u_i}{A_u} + \frac{I_i(x,y) - v_i}{A_v}\right), \quad i = 1, 2, \tag{21}$$

where A_u and A_v are the areas inside and outside the contour, respectively. Combining these forces using the weights $\{w_i(x, y)\}$, i=1,2, and adding the penalty term on the total arc length of the curve $\vec{C}$, the following optimal flow is obtained:

$$\frac{d\vec{C}(x,y)}{dt} = \sum_{i=1}^{2} w_i(x,y) F_i(x,y) \vec{N}_{out} - \alpha\kappa\vec{N}_{out}, \tag{22}$$

where $\vec{N}_{out}$ is the outward unit normal of $\vec{C}$ at (x, y), $\alpha \in [0, 1]$ is a constant weight on the total arc length, and κ is the curvature.

The initial contour is shown in Figure 31. The results of the three schemes are displayed in the pseudo-color images (the last three columns) in Figure 32. They are compared to the monochannel segmentation results on the first and second channels, respectively, which are shown in Figure 32 as the gray-scale images in the first two columns. The results shown were obtained after 1500 iterations of evolution with a fixed step size of 0.05 and α of 0.

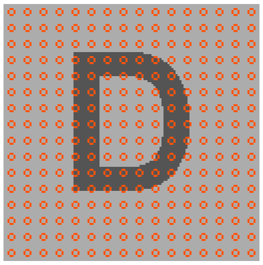
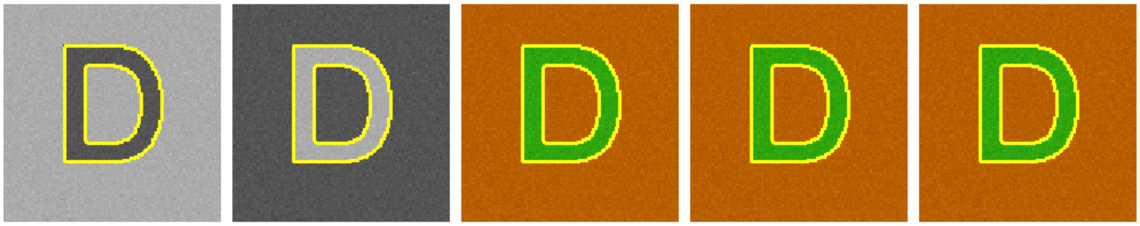


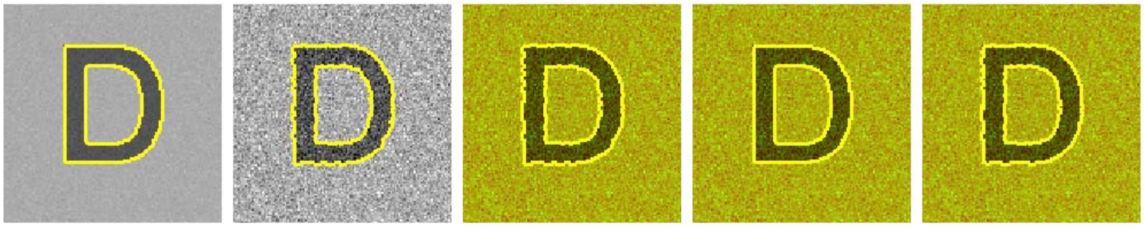
Figure 31. The initial contour on test images.

(1) Reversed intensities ($u_1=v_2$, $v_1=u_2$, $\sigma_1 = \sigma_2 = 5$)



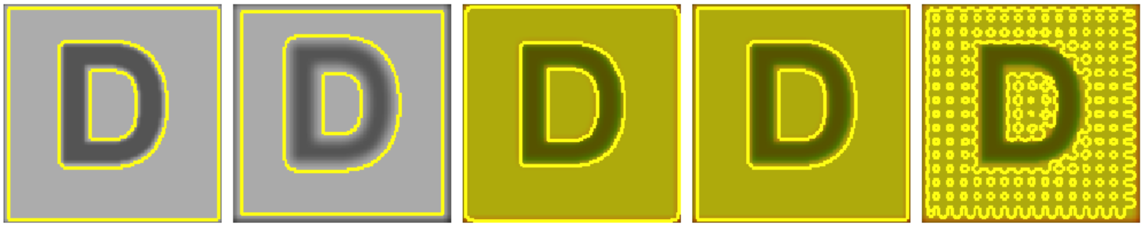
Channel 1 Channel 2 $w_1=w_2=0.5$ $w_1=0.50$, $w_2=0.50$ Local weight

(2) Different noise powers ($\sigma_1 = 5$, $\sigma_2 = 30$)



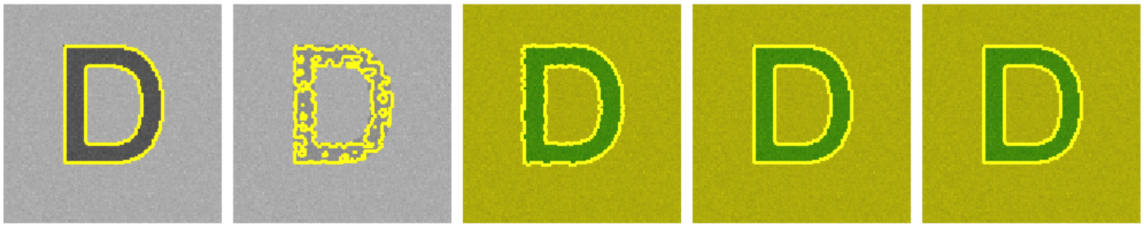
Channel 1 Channel 2 $w_1=w_2=0.5$ $w_1=1.00$, $w_2=0.00$ Local weight

(3) Different blurring (Gaussian blur, chl 1: 5x5 mask with $\sigma = 1$; chl 2: 10x10 mask with $\sigma = 3$)



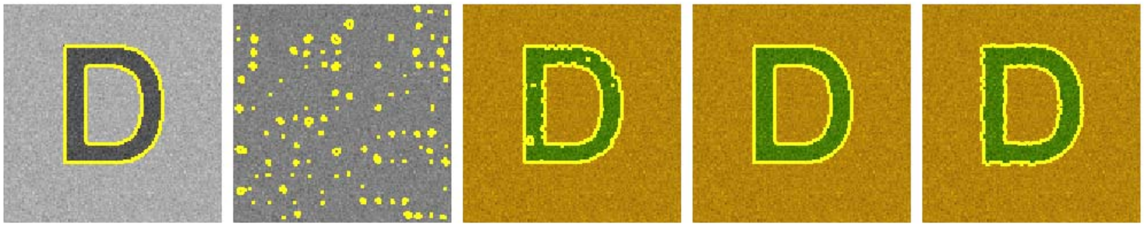
Channel 1 Channel 2 $w_1=w_2=0.5$ $w_1=1.00$, $w_2=0.00$ Local weight

(4) Different contrast (u_2-v_2 = (u_1-v_1)/3, $\sigma_1 = \sigma_2 = 5$)



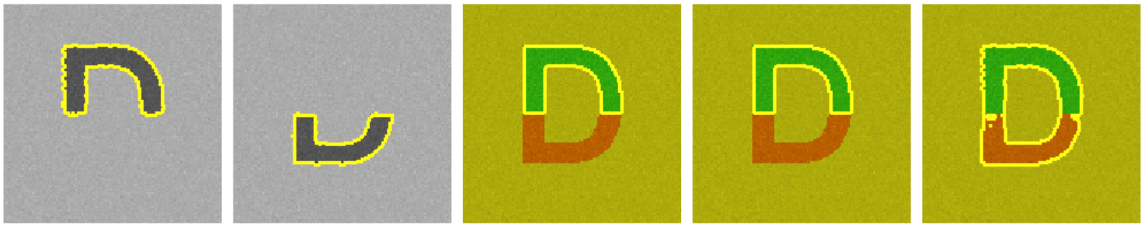
Channel 1 Channel 2 $w_1=w_2=0.5$ $w_1=1.00$, $w_2=0.00$ Local weight

(5) Only one channel shows the feature ($\sigma_1 = \sigma_2 = 10$)



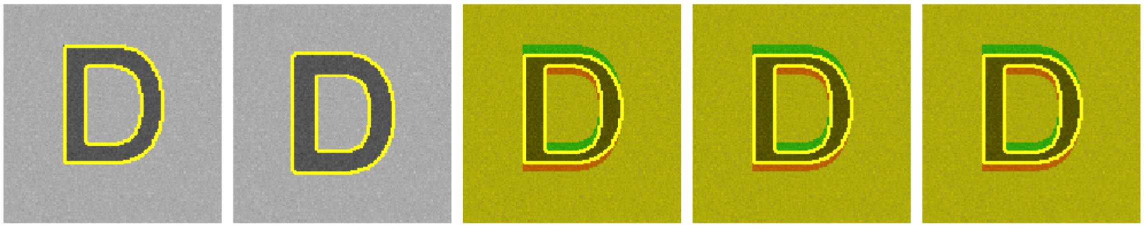
Channel 1 | Channel 2 | $w_1=w_2=0.5$ | $w_1=0.99, w_2=0.01$ | Local weight

(6) Channels show complementary features ($\sigma_1 = \sigma_2 = 5$)



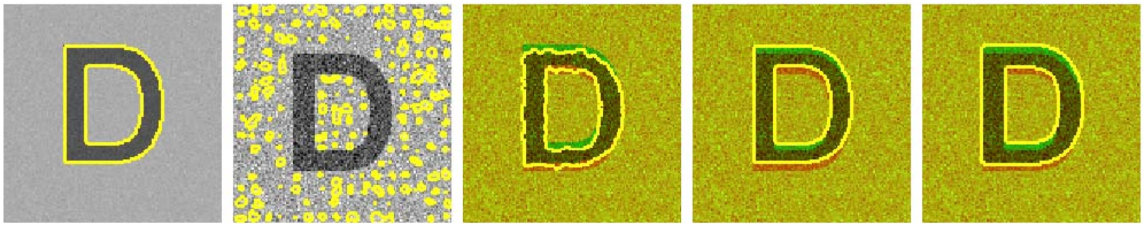
Channel 1 | Channel 2 | $w_1=w_2=0.5$ | $w_1=0.61, w_2=0.39$ | Local weight

(7) Registration error with equal noise powers ($\sigma_1 = \sigma_2 = 5$)



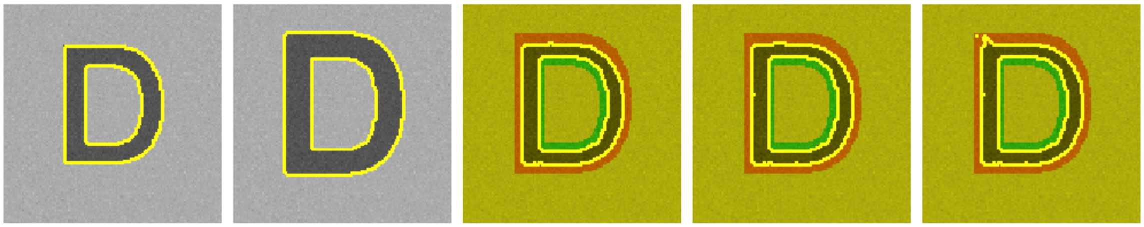
Channel 1 | Channel 2 | $w_1=w_2=0.5$ | $w_1=0.50, w_2=0.50$ | Local weight

(8) Registration error with unequal noise powers ($\sigma_1 = 5$, $\sigma_2 = 30$)



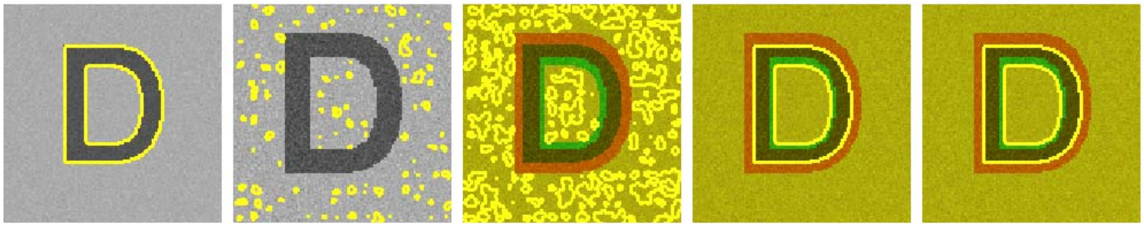
Channel 1 | Channel 2 | $w_1=w_2=0.5$ | $w_1=1.00, w_2=0.00$ | Local weight

(9) Contradictory features with equal noise powers ($\sigma_1 = \sigma_2 = 5$)



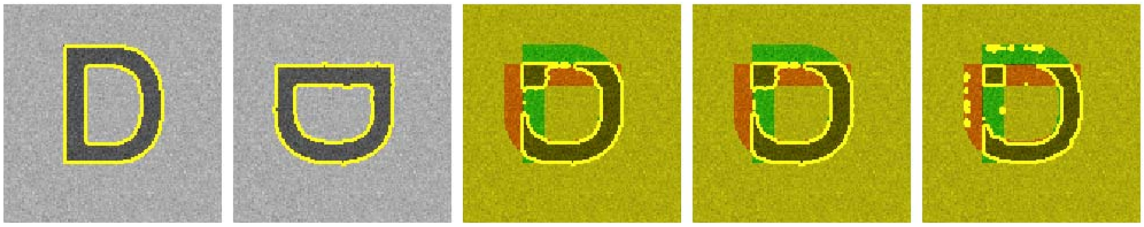
Channel 1 | Channel 2 | $w_1=w_2=0.5$ | $w_1=0.43$, $w_2=0.57$ | Local weight

(10) Contradictory features with unequal noise powers ($\sigma_1 = 5$, $\sigma_2 = 10$)



Channel 1 | Channel 2 | $w_1=w_2=0.5$ | $w_1=0.98$, $w_2=0.02$ | Local weight

(11) Two channels show different features ($\sigma_1 = \sigma_2 = 10$)



Channel 1 | Channel 2 | $w_1=w_2=0.5$ | $w_1=0.50$, $w_2=0.50$ | Local weight

Figure 32. Results of monochannel and multichannel segmentations on test images. Columns 1 and 2: monochannel segmentation on channels 1 and 2; Columns 3 to 5: multichannel segmentation using constant equal weights (CEW), global constant weights (GCW), and window-based local weights (WLW).

From the above results, the following observations can be made:

(a) For redundant and compatible feature, i.e., the edges occur in both channels at the same positions, as in Figure 32 (1) to (4), all three schemes are able to find the accurate edges, despite the different weights applied.

(b) For complementary features, i.e., the edge segments that show in one channel do not exist in the other, as in (5) and (6) in Figure 32, different weighting schemes result in different segments. In (5), the force in channel 1 is greater because of the existence of the letter “D”. Therefore, all schemes captured the boundaries. In (6), the letter “D” is cut into two non-overlapping pieces and each channel shows only one piece. The first two schemes captured the piece in channel 1 only, while the WLW scheme was able to capture features in both channels. The reason is that channel 1 has a larger portion of “D”, which results in a force stronger than the other, i.e., $\left|\vec{F}_1\right| > \left|\vec{F}_2\right|$. This in turn yields a combined force in the direction of $\vec{F}_1$ when CEW or GCW are used, and the contours in the lower portion of “D” eventually disappeared. In contrast, the WLW scheme combines the two force components according to the local weights shown in Figure 33, where much higher weights are put on the channel at the locations the feature can be observed. As a result, the boundaries in both channels were finally successfully located.

(c) For contradictory features, i.e. the edge features appear in both channels but there is an apparent displacement because of either registration error in (7) and (8) or different sensor characteristics in (9) to (11), the way how contradictions is resolved is different with schemes. When feature areas are comparable and the noise powers are equal, the weights are approximately equal for all schemes and the snakes stopped at the intersection's boundaries ((7), (9), and (11)). When the noise powers are not equal ((8) and (10)), the snakes evolve differently with schemes. The CEW method is slower than the other two methods simply because the forces are in opposite directions and the averaged force is weaker. The other two methods, on the other hand, put greater weights on the channel with less noise and finally capture the feature in that channel, with a much faster convergence speed than that of the CEW method.

The results on test images illustrated that the weight selection is not a trivial issue, especially when channels demonstrate features that are complementary or contradictory. When channels show complementary features, the WLW method is effective in capturing all features. When contradiction arises, both the GCW and WLW methods correctly identify the more “trustful” sources and then quickly locate the reasonable boundaries.

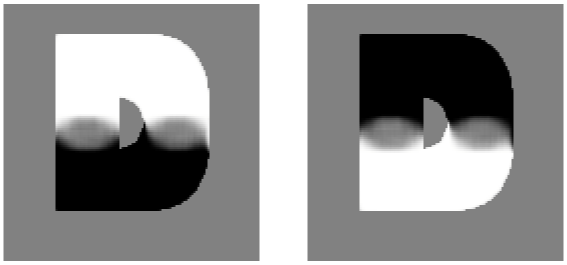


Figure 33. The weights on channels 1 and 2 in Figure 32 (6), respectively, when using the WLW method.

4.4.2 Experiment on Real Images

Experiments were carried out on three pairs of real images and the results are shown in Figures 34 to 37. The images are

- the microarray images generated in gene expression studies of skeletal muscle repair after traumatic or strain injury (Figure 34),
- the MR and ultrasonic scan images of a baby's brain[2] (Figure 35), and

---

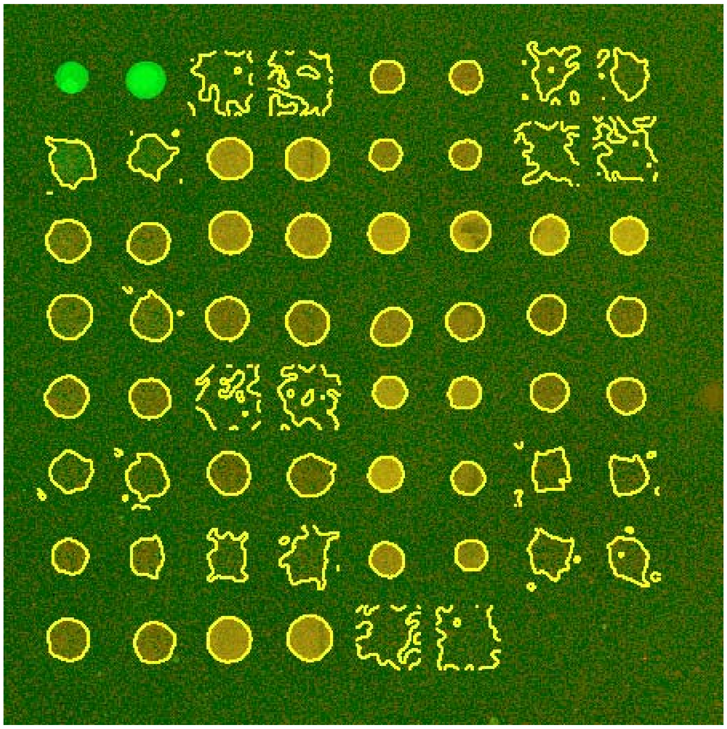
[2] Images obtained from http://www-sop.inria.fr/epidaure/Demonstrations/roboscope/index.htm

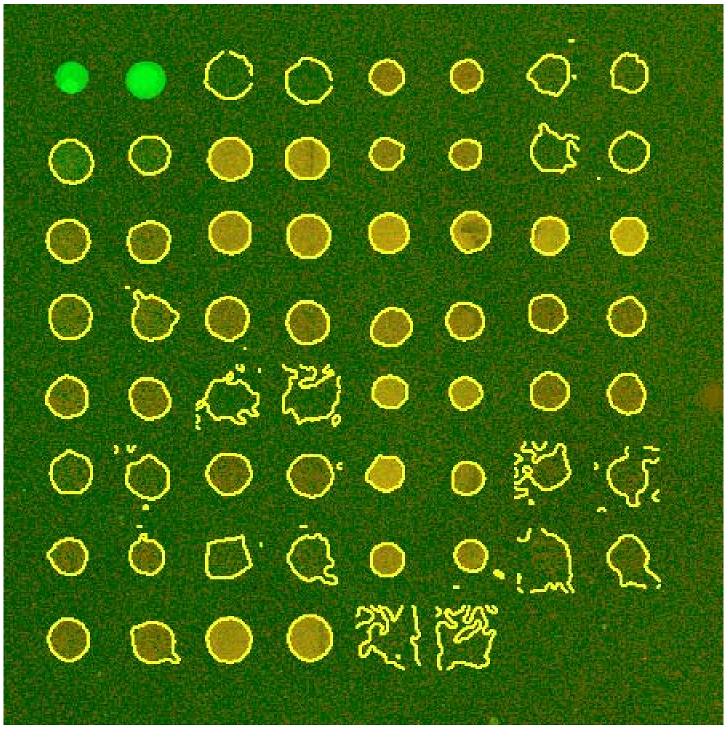
- the visible band blue and infrared images of New York City[3] (Figure 36 and Figure 37).

In the microarray image example, the goal is to separate the circular features from the background. Two images are obtained by scanning two slides of the same gene matrices. The original images show very poor contrast and high noise. Therefore, before the images are processed, histogram equalization and Wiener filtering were applied to expand the dynamic range and to decrease the noise. Then segmentation is performed on the neighborhood of each feature so that each feature is segmented independently. As shown in the first two images in Figure 34, some circular features have very poor contrast with the background, and monochannel segmentation on those features was not satisfactory. By using the proposed multisensory snake method with the local weights, we expect the snake to select the more accurate feature boundaries. The last image in Figure 34 illustrates that the algorithm achieved this goal. For example, the four features in the upper right corner were poorly segmented in channel 1 monosensory segmentation but successfully segmented by the proposed method. Even some of the features that are failed by both channels have been located successfully (e.g. the 4th and 8th features on the 7th row.)

---

[3] Images courtesy of Dr. John C. Russ, who is a visiting professor in Material Science and Engineering Department, North Carolina State University.





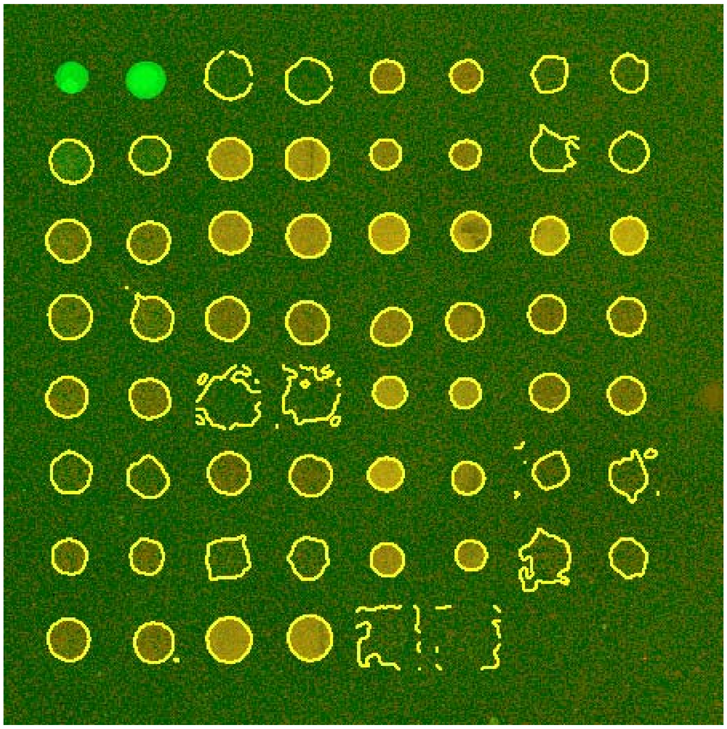


Figure 34. Segmentation of microarray images generated in gene expression studies. After 50 iterations with a step size of 0.2. Top and middle: monochannel segmentation on channels 1 and 2, respectively; Bottom: segmentation using the developed algorithm.

In the example of baby brain images of MRI and ultrasonic scan (US) (Figure 35), monosensory segmentation was either unsatisfactory or slightly different across channels. Before applying the proposed fusion scheme, smoothing filter is applied to both images to avoid giving too much weight on the MRI image where it has tiny features. The calculated weights on individual images are displayed as gray-scale images in the third and fourth columns. It can be seen that at features boundaries, the MRI image was given much higher weights, which resulted in more accurate segmentation.

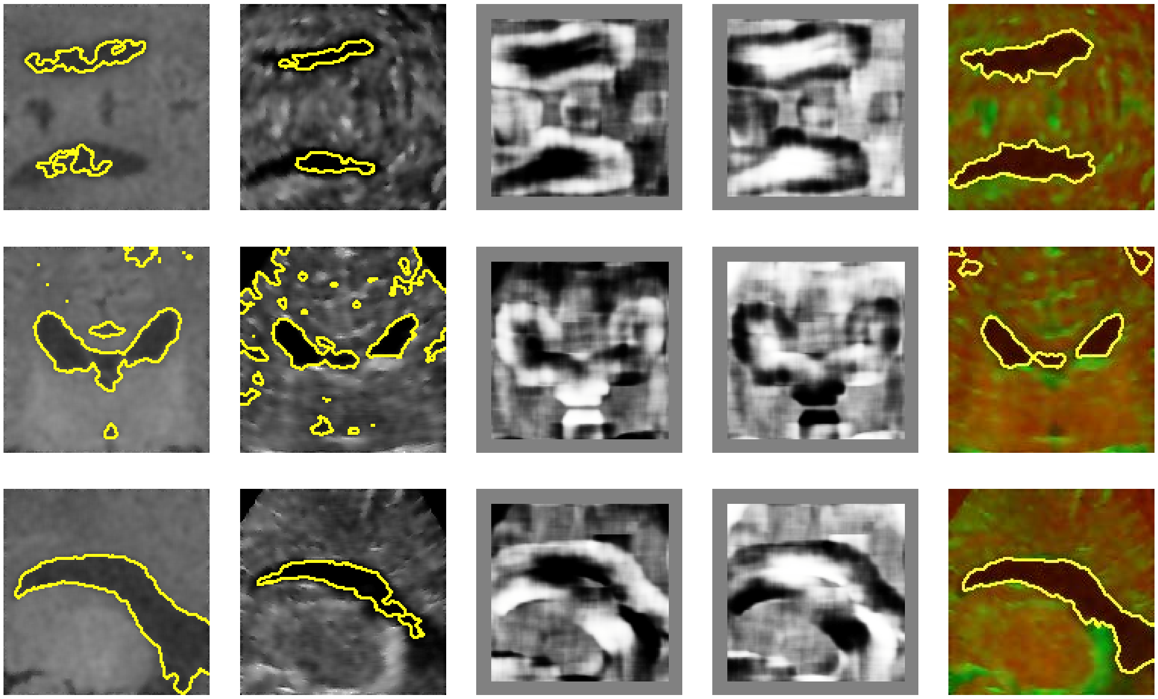


Figure 35. Segmentation of a baby's brain MR and ultrasonic scan (US) images. After 1000 iterations with a step size of 0.1. First two columns: monosensory segmentation of the filtered MR images and US images, respectively; 3rd and 4th columns: local weights obtained on the MR and US images, respectively; the last column: multisensory segmentation result.

The last example is a pair of registered LANDSAT images of New York City. Better contrast is observed in the IR image than in the visible blue-band image. The cyan circles highlighted some features that can be observed in the IR channel but not in the blue one. The segmentation results using the proposed approach, as shown in the last clip in Figure 37, clearly illustrate that boundaries of the rivers were accurately located using the local weights computed from M_{FDR}, even for all the contradictory features in the circles.

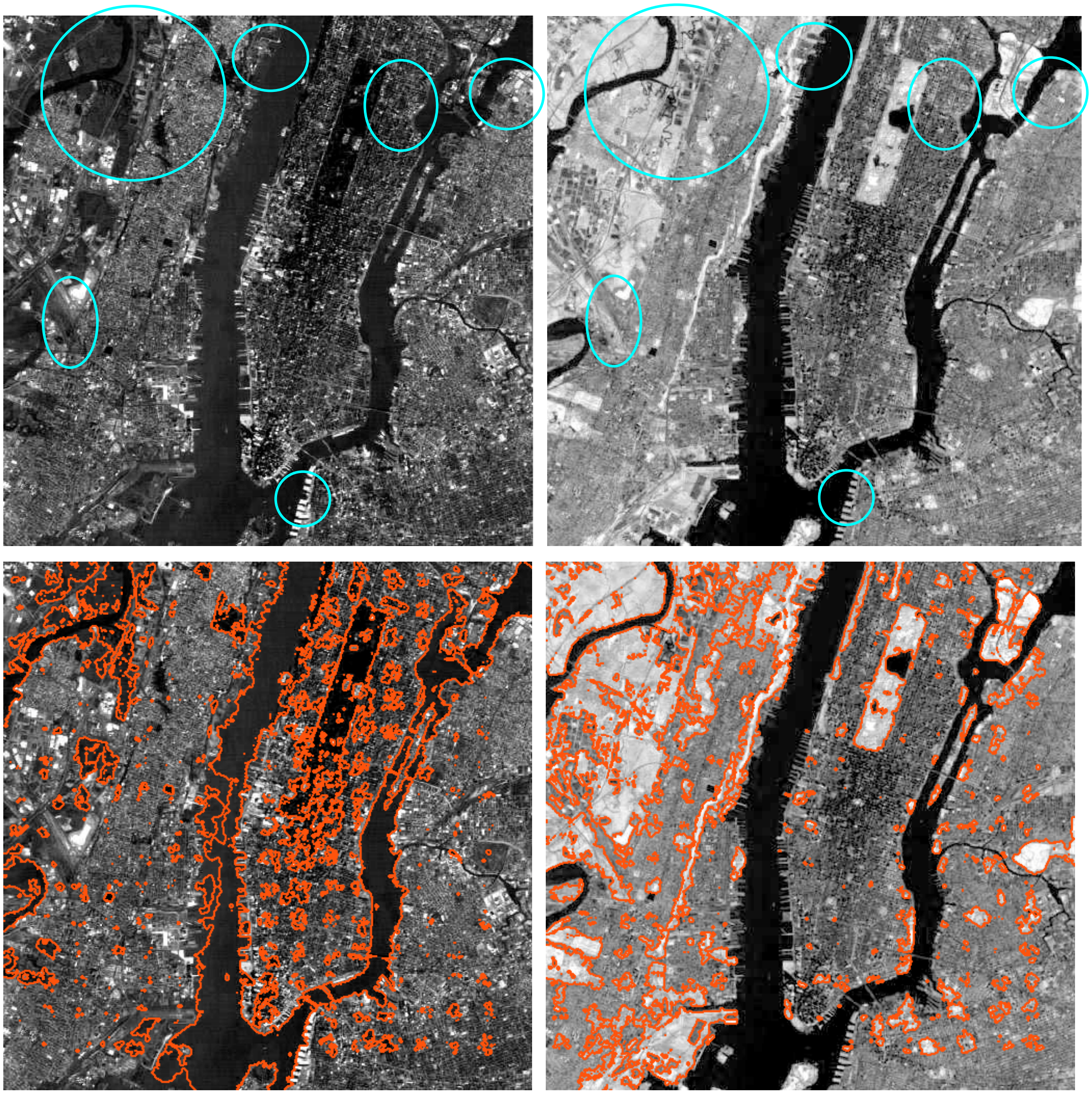


Figure 36. Monosensory segmentation of the aerial images of visible blue band (left) and IR (right) of New York City after 2000 iterations with a step size of 0.1. Top: original images. Bottom: the images showing the final snakes.

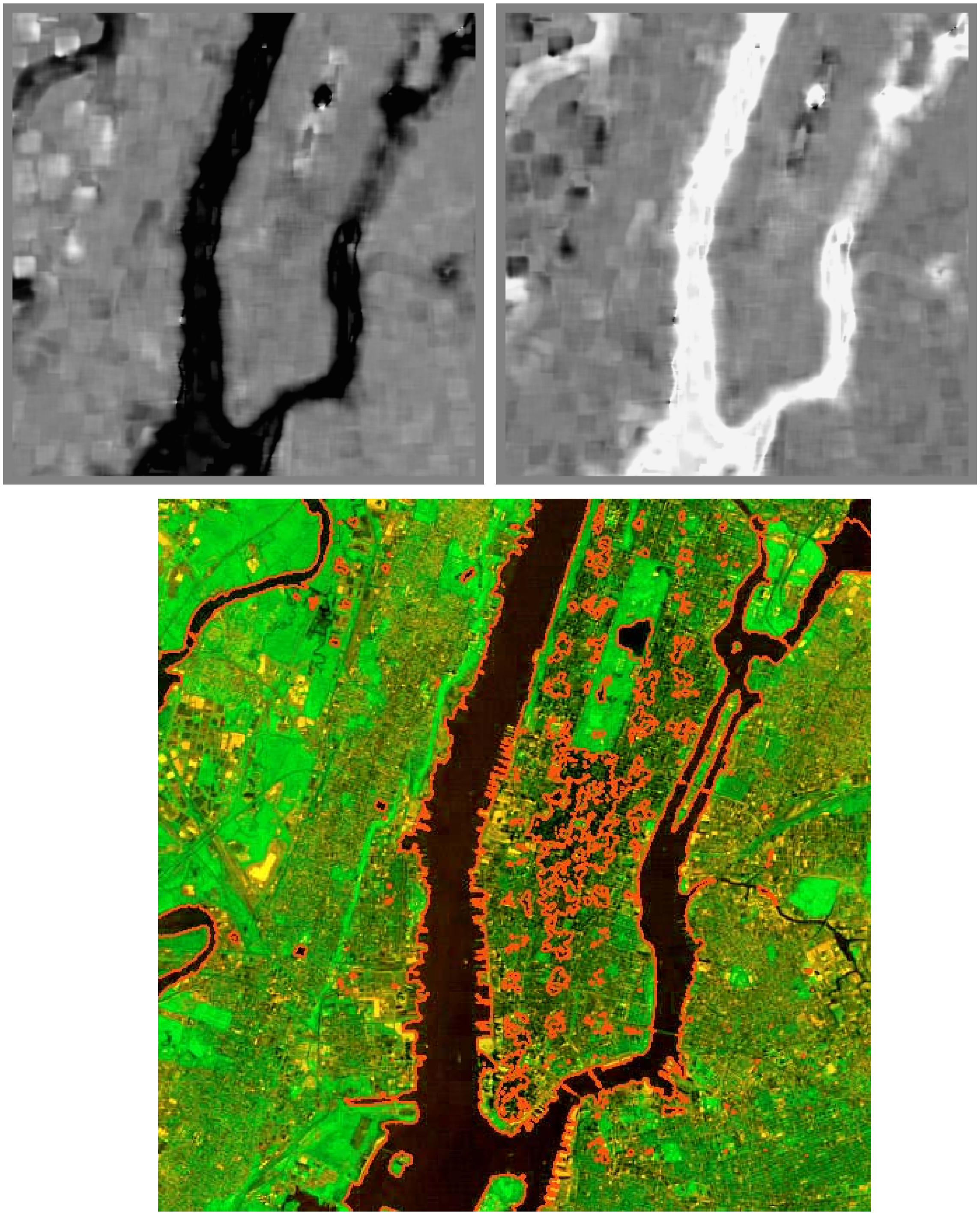


Figure 37. Multisensory segmentation after 2000 iterations with a step size of 0.1. Top: weights on the blue and IR images, respectively. Bottom: Segmentation using the developed algorithm with WLW.

4.5 Segmentation of X-Ray and Color Images

The proposed method is employed to segment x-ray and color images of bone-contaminated poultry breast. Fan bones are barely visible to x-ray since it contains only a small amount of calcium and is very thin. They are generally a little darker in the image than the surrounding meat, whose intensity changes slowly because of varying meat thickness. Therefore, the segmentation of fan bones on x-ray images is a challenging task. In the mean time, although the segmentation on the visible RGB image is relatively easier and more accurate, it does produce significant under-segmentation errors, especially when illumination is not uniform enough. In addition, it also generates a large amount of false regions of dark meat that look similar in color and shading to bones. By combining two modalities in segmentation, we expect to achieve more accurate segmentation and also to reduce the number of false regions, so that the classification is easier and eventually the overall system performance is improved.

An example of the registered images is shown in Figure 38, where (a) and (b) are the clips of the visible band and x-ray images, respectively, for the same fan bone-contaminated chicken part. The fan bone is marked out with a red circles in both images. The x-ray image and the red channel of the visual image are overlapped to form a pseudo-color image in (c). Significant registration errors (misalignment between the transformed x-ray image and the visual image) are observed in the registered images. To show this, the boundaries of the same fan bone in individual images are traced manually, where the green curve corresponds to the x-ray image and the red curve to the visual image. The main reason of the registration error is that the assumption of linear conformal transform between the two images is not totally valid. Besides, error is introduced during manual selection of control points because of distinct sensor sensitivities: one common subject (such as a fan bone) may have distinct views in images of different modalities. Therefore, segmentation algorithms that can accommodate the registration error need to be employed.

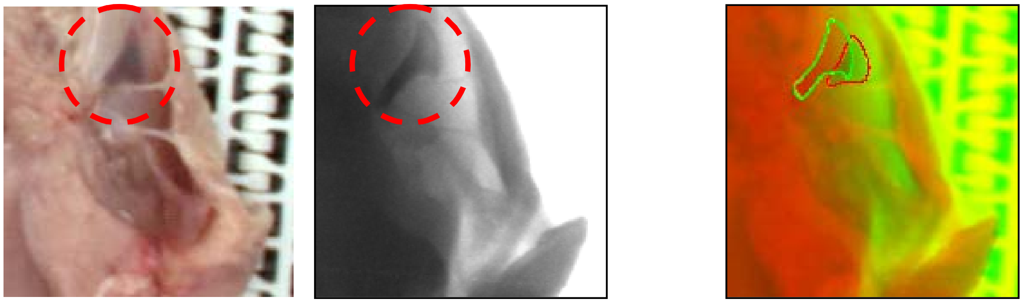

(a) visual image (b) x-ray image (c) pseudo-color image with fan bone boundaries marked out

Figure 38. A pair of registered visible color and x-ray images with registration error.

Prior to segmentation, both the color and the x-ray images are preprocessed to enhance the contrast using gray-scale morphological operations. The initial contour is obtained by thresholding both the red and the x-ray channels. The preprocessed images and the snakes resulted from monochannel segmentation for individual channels are shown in Figure 39. The segmentation of the x-ray image completely failed to catch the fan bone.

The combined-force algorithm with local weights resulted in the snakes displayed in Figure 40. Because of the significant registration error between the two images and the heavier weights on the visible red channel, the resulting snake was misplaced by an offset from the fan bone boundary in the x-ray image. This error needs to be carefully interpreted at higher levels of processing. In our performance assessment, we do not make distinguish on which channel's feature the snake is capturing in presence of registration errors.

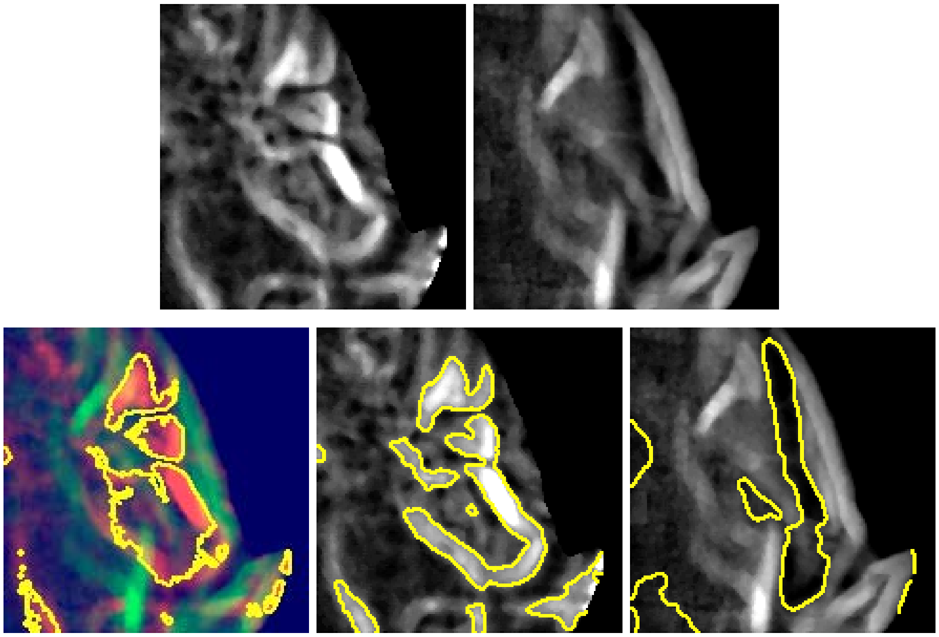


Figure 39. Filtered red band (upper left) and x-ray (upper right) images and their monosensory segmentation results at the bottom.
Bottom left: initial contour; middle and right: snakes after 500 iterations on the filtered visible-band red image and x-ray image, respectively.

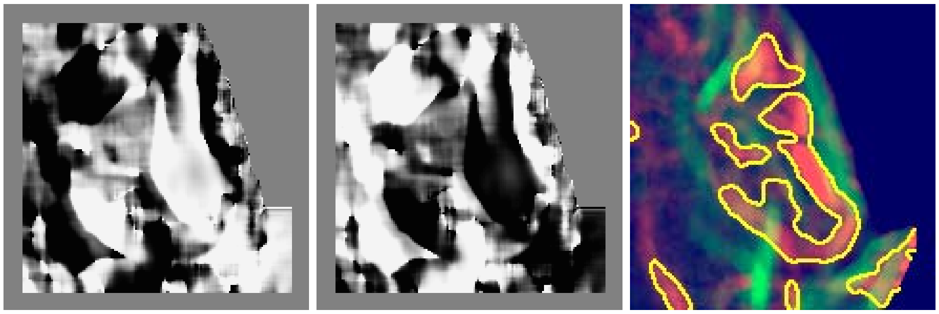


Figure 40. The multisensory segmentation after 500 iterations (step size = 0.1, and $\alpha = 0.2$.) Left and middle: local weight on the visible red and the x-ray channels, respectively; right: multisensory segmentation results.

Another example is shown in Figure 41. It can be seen that apart from more accurate segmentation of the fan bone regions, the snakes also reduced the false regions generated in the visible red channel segmentation significantly. Besides, the remaining one false region was deformed significantly and thus easier to classify.

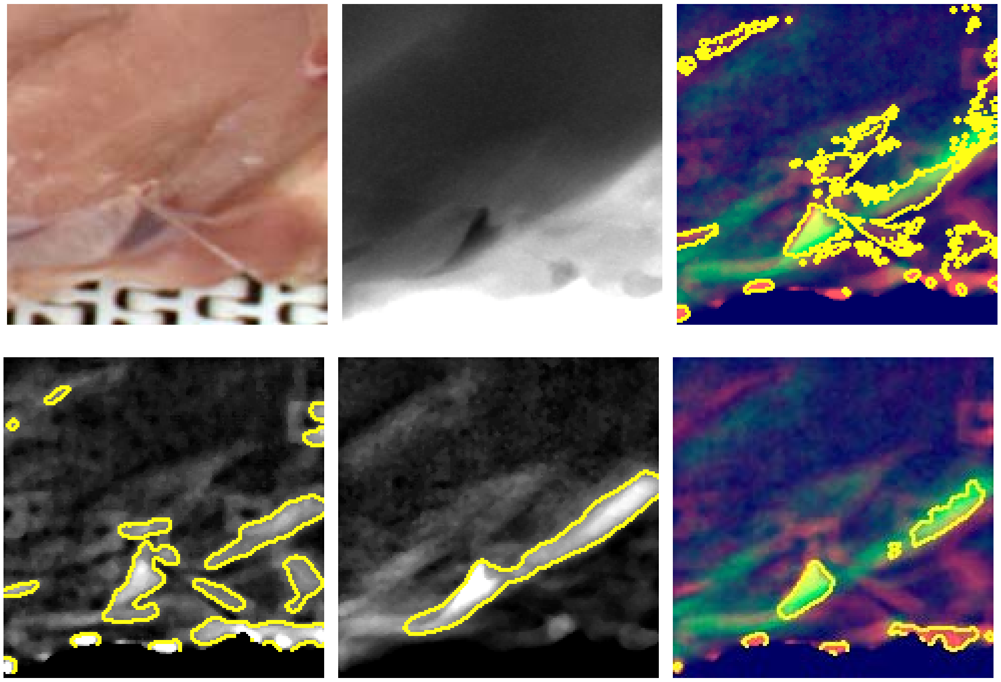


Figure 41. Another example of multisensory fan bone segmentation.
From left to right and top to bottom: original color image, original x-ray image, initialized snakes, monosensory red channel segmentation, monosensory x-ray segmentation, and multisensory segmentation with local weights.

A small number of images from the chicken image database are color and x-ray pairs taken on-line in a poultry plant. They were used to test the proposed multisensory segmentation algorithm. Totally 51 fan bone-contaminated image clip pairs of the size 160×160 were cut from the registered visual and x-ray images. Visual assessment was carried out on both the monochannel and multisensory segmentation results, as listed in Table 6.

Table 7 provides the more detailed performance numbers of the red channel segmentation versus the multisensory WLW method. It can be seen that the developed scheme achieved 37 good segmentations, which is a significant improvement over the visible red results of 28. In particular, out of the 18 clips that the monosensory segmentation in the red channel totally failed, the WLW method was able to correct 7 successfully. In the mean time, out of the 28 clips that the monosensory results were good, only 1 was badly failed by the WLW method. This improvement verified that the snakes with the local weights calculated from M_{FDR} have the capability of balancing different forces and seeking the right way toward the true boundary, in spite of the significant registration errors. The snakes using CEW achieved even less good segmentations than the monosensory red channel does. This confirmed that the weight computation method is critical and is not a trivial problem.

Table 6. Result comparison of monosensory and multisensory segmentation algorithms.

Segmentation Algorithms		***Good***	***Fair***	***Poor***
Monosensory	***Visible Red***	28	5	18
	X-Ray	13	9	29
Multisensory	***Constant Equal Weights* (*CEW*)**	25	8	18
	***Local Weights* (*WLW*)**	37	4	10

Table 7. Result comparison of visible red monosensory and multisensory WLW methods.

WLW ***Red***	***Good***	***Fair***	***Poor***	**Total**
Good	25	2	1	28
Fair	5	0	0	5
Poor	7	2	9	18
Total	37	4	10	51

The above results illustrated that the developed algorithm is effective in improving the segmentation accuracy and capable of reducing the number of false regions and making them easier to interpret.

4.6 Bone Inspection Based on Color and X-Ray Images

Based on the developed multisensory segmentation algorithm, we propose a fan bone detection algorithm to effectively combine the information from visible band and x-ray image modalities. With the flow chart depicted in Figure 42, it consists of the following steps:

(a) The position and orientation of the chicken part are computed from the x-ray image.

(b) Color analysis is performed using the visible band image to obtain possible positions of fan bones. Then the visual and the x-ray images are clipped accordingly.

(c) Gray-scale morphological filtering is applied to both the x-ray and visible red image clips.

(d) The snake algorithm is applied to the filtered image clips. The contour is initialized by thresholding both channels.

(e) For each blob generated by the segmentation, numerical features are extracted. Note that the spectral features are computed for both images.

(f) PNN classifier is deployed to assign an identity to each region.

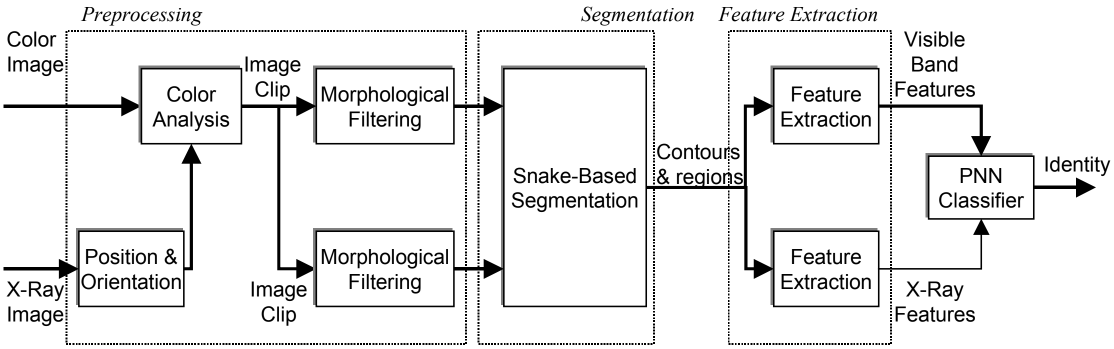


Figure 42. Inspection algorithm based on the serialized segmentation scheme.

CHAPTER 5

CONCLUSIONS AND CONTRIBUTIONS

Although vision-based inspection is not new, it was made possible only within the past two decades for real-time, on-line, and 100 percent inspection of high-volume product, thanks to the emerging advanced technologies in sensing, computing, signal and image processing, and pattern recognition. The objective of this research is to develop a general methodology for real-time vision-based inspection with a focus on advanced image processing, fusion, and analysis techniques.

The first part of this dissertation describes a general methodology for real-time vision-based inspection. It includes an image acquisition module with diffusive lighting to maximize the contrast and eliminate the artifacts, a preprocessing module to further enhance the contrast and remove irrelevant information, a snake algorithm for fast and accurate segmentation, a feature extraction module, and a classification module using a PNN classifier. A recursive segmentation and classification scheme is adopted so that the segmentation results that are not accurate can be sent back to the segmentation module to be corrected. The segmentation accuracy is determined using the output confidence level of the PNN classifier. This methodology is applied in the real-time bone inspection system for deboned meat and achieved a detection rate of 92.3% and a false alarm rate of 7.0% from the lab-scale testing of 280 samples.

In the second part of the dissertation, a fast snake-based multisensory image segmentation algorithm is presented for the purpose of inspection using multiple sensors.

In this algorithm, the snakes are driven by the weighted sum of the optimal forces derived from corresponding energy functionals in each channel. The weights are calculated based on a novel metric that measures the local contrast and noise power in individual sensor images. The developed algorithm is tested on both artificially created images and real images, and reasonable results are obtained. The results on visible band color and x-ray images of the deboned meat illustrate that, in addition to higher segmentation accuracy than the monosensory results, the developed algorithm tolerates a considerable amount of registration error.

In this research, some general guidelines for vision-based inspection algorithm design were pointed out. One of the future research topics will be to investigate more such guidelines in selecting appropriate algorithms to expedite the design process. When testing the bone detection system, we found that even the smallest difference in factory set up may require considerable parameter tuning of the algorithms. Therefore, advanced techniques such as color normalization methods are necessary to increase the robustness to variations in lighting, samples, and defects.

Related to multisensory image segmentation using snakes, one future research topic is to investigate new weight computation methods that do not require histogram matching of different images. Also the stability and convergence of the proposed multisensory snakes need to be verified. The interpretation of segmentation results, especially in presence of registration error, is a challenging topic because the segmentation results may be interpretable in only some of the channels.

This research has lead to the following major contributions:

- Proposed a general methodology for real-time vision-based inspection of abnormality that contains certain shape and spectral patterns.
- Developed and tested a recursive segmentation and classification scheme for real-time segmentation with high accuracy.
- Applied the developed inspection methodology successfully to fan bone detection of deboned poultry meat and achieved satisfactory results.
- Derived and verified a fast snake-based multisensory image segmentation algorithm and applied it to the segmentation of visible band and x-ray images of deboned meat.

REFERENCE

[1] P. Bonnin, B. Hoeltzener-Douarin, and E. Pissaloux, “A New Way of Image Data Fusion: The Multi-spectral Cooperative Segmentation,” *Proc. Intl. Conf. Image Processing*, vol.3, 1995, pp. 572 –575.

[2] K. L. Boyer and T. Ozguner, “Robust Online Detection of Pipeline Corrosion from Range Data,” *Machine Vision and Applications*, no. 12, pp. 291–304, 2001.

[3] P. J. Burt, and R. J. Kolczynski, “Enhanced Image Capture Through Fusion,” *Proc. 4th Intl. Conf. Computer Vision*, 1993, pp. 173-182.

[4] V. Caselles, R. Kimmel, and G. Sapiro, “Geodesic Active Contours,” *Intl. J. Computer Vision*, vol. 22, no. 1, pp. 61-79, 1997.

[5] R. Caves, S. Quegan, and R. White, “Quantitative Comparison of the Performance of SAR Segmentation Algorithms,” *IEEE Trans. Image Processing*, vol. 7, no. 11, pp. 1534-1546, Nov. 1998.

[6] R. M. Cesar Jr. and L. F. Costa, “Shape Analysis and Classification: Theory and Practice,” Boca Raton, FL: CRC Press LCC, 2000.

[7] T. F. Chan, B. Y. Sandberg, and L. A. Vese, “Active Contours without Edges for Vector-Valued Images,” *J. Visual Communication and Image Representation*, vol. 11, pp. 130-141, 2000.

[8] J. Chanussot, G. Mauris, and P. Lambert, “Fuzzy Fusion Techniques for Linear Features Detection in Multitemporal SAR Images,” *IEEE Trans. Geoscience and Remote Sensing*, vol. 37, no. 3, part 1, pp. 1292 –1305, May 1999.

[9] B. Charroux, S. Phillipp, and J-P. Cocquerez, “Image Analysis: Segmentation Operator Cooperation Led by the Interpretation,” *Proc. Intl. Conf. Image Processing*, vol. 3, pp. 939 - 942, 1996.

[10] Y. -R. Chen, K. Chao, and M. Kim, “Machine Vision Technology for Agricultural Applications,” *Computers and Electronics in Agriculture*, pp. 1-19, 2002.

[11] E. R. Davies, “Algorithms for Inspection: Constraints, Tradeoffs and the Design Process,” *IEE Colloquium on Industrial Inspection* (Digest No: 1997/041), pp. 6/1 - 6/5, 1997.

[12] E. R. Davies and D. Patel, “Sensitive X-Ray Detection of Contaminants in Food Products,” *IEE Colloquium on Application of Machine Vision*, pp. 2/1-2/6, 1995.

[13] Y. Ding, A. Yezzi, B. Heck, W. Daley, G. Vachtsevanos, and Y. Zhang, "An On-line Real-time Automatic Visual Inspection Algorithm for Surface Bone Detection in Poultry Products," 2nd *WSEAS Intl. Conf. Signal, Speech, and Image Processing* (WSEAS ICOSSIP 2002), Koukounaries, Skiathos Island, Greece, September 25-28, 2002.

[14] Y. Ding, G. Vachtsevanos, A.Yezzi, Y. Zhang, and Y. Wardi, "A Recursive Segmentation and Classification Scheme for Improving Segmentation Accuracy and Detection Rate in Real-time Machine Vision Applications," *Proc. 14th Intl. Conf. Digital Signal Processing* (DSP2002), vol. II, pp. 1009-1014, Santorini, Greece, July 1-3, 2002.

[15] R. K. Falah, P. Bolon and J.P. Cocquerez, "A Region-Region and Region-Edge Cooperative Approach of Image Segmentation," *1st IEEE Intl. Conf. Image Processing*, vol. 3, pp. 470-474, 1994.

[16] L. Fatone, P. Maponi, and F. Zirilli, "Fusion of SAR/Optical Image to Detect Urban Areas," *IEEE/ISPRS Joint Workshop on Remote Sensing and Data Fusion over Urban Areas*, pp. 217 – 221, 2001.

[17] C. Fiorio and R. Nock, "Image Segmentation Using a Generic, Fast and Non-Parametric Approach," *Proc. 10th IEEE Intl. Conf. Tools with Artificial Intelligence*, pp. 450-458, 1998.

[18] M. Graves, A. Smith, and B. Batchelor, "Approaches to Foreign Body Detection in Foods," *Trends in Food Science and Technology*, Elsevier 9, pp.21-27, 1998.

[19] D. L. Hall, "Mathematical Techniques in Multisensor Data Fusion," Norwood, MA: Artech House, 1992.

[20] D. L. Hall and J. Llinas, "Handbook of Multisensor Data Fusion," CRC Press LLC, 2001.

[21] S. L. Hegarat-Mascle, I. Bloch and D. Vidal-Madjar, "Application of Dempster-Shafer Evidence Theory to Unsupervised Classification in Multisource Remote Sensing," *IEEE Trans. Geoscience and Remote Sensing*, vol. 35, no. 4, pp. 1018-1031, July 1997.

[22] F. Huet and S. Philipp, "Fusion of Images after Segmentation by Various Operators and Interpretation by a Multi-Scale Fuzzy Classification," *Proc. 14th Intl. Conf. Pattern Recognition*, vol.2, pp. 1843 –1845, 1998.

[23] N. Ikonomakis, K. N. Plataniotis, M. Zervalis, and A. N. Venetsanopoulos, "Region Growing and Region Merging Image Segmentation," *Proc. Intl. Conf. Digital Signal Processing* (DSP-97), vol. I, pp. 299-303, July 2-4, 1997.

[24] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active Contour Models, " *Intl. J. Computer Vision*, pp. 321-331, 1988.

[25] G. Koepfler, C. Lopez, and L. Rudin, "Data Fusion by Segmentation. Application to Texture Discrimination," *Actes du 14*me *Colloque GRETSI*, pp.707-710, Juan-les-Pins, Sept. 1993.

[26] Y. Li, Y. V. Venkatesh, and K. C. Chung, "Multisensor Image Fusion Using Influence Factor Modification and the ANOVA Methods," *IEEE Trans. Geoscience and Remote Sensing*, vol. 38, no. 4, pp.1976 - 1988, July 2000.

[27] D. Patel, I. Hannah and E. R. Davies, "Automatic Detection of Foreign Objects in Food Products," *4th Intl. Conf. Factory 2000 – Advanced Factory Automation*, pp. 517 – 522, Oct. 1994.

[28] E. Peli, "Feature Detection Algorithm Based on a Visual System Model," *Proceedings of the IEEE*, vol. 90, no. 1, pp. 78 –93, Jan. 2002.

[29] R. Pye, "Vision Inspection: Meeting the Promise?" *Proc. Intl. Test Conference*, pp. 1035, 1997.

[30] A. R. Rao, "Future Directions in Industrial Machine Vision: A Case Study of Semiconductor Manufacturing Applications," *Image and Vision Computing*, vol. 14, pp. 3-19, 1996.

[31] G. Sapiro, "Color Snakes," *Hewlett-Packard Labs Technical Report 113*, Sept. 1995.

[32] G. Seetharaman and C. H. Chu, "Image Segmentation by Multisensor Data-Fusion," *22nd Southeastern Symposium on System Theory*, pp. 583 –587, 1990.

[33] J. A. Sethian, "Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Science," Cambridge, UK: Cambridge University Press, 1996.

[34] J. Shah, "Curve Evolution and Segmentation Functionals: Application to Color Images," *Proc. IEEE Intl. Conf. Image Processing*, vol. I, pp. 461-464, Sept.1996.

[35] D. Smith, "Bones in Boneless Broiler Breast Meat Is a Legitimate Concern," *World Poultry*, vol.15, pp. 35-36, 1999.

[36] M. Smith, "A Study Guide for Digital Image Processing," Riverdale, GA: Scientific Publishers, 1999.

[37] B. Solaiman, R. K. Koffi, M-C Mouchot, and A. Hillion, "An Information Fusion Method for Multispectral Image Classification Postprocessing," *IEEE Trans. Geoscience and Remote Sensing*, vol. 36, no. 2, pp. 395 – 406, Mar. 1998.

[38] D. F. Specht, "Probabilistic Neural Networks for Classification, Mapping, or Associative Memory," *Proc. IEEE Intl. Conf. Neural Networks*, vol. 1, pp. 525-532, June 1988.

[39] C. Spinu, C. Garbay, and J. M. Chassery, "A Multi-Agent Approach to Edge Detection as a Distributed Optimization Problem," *Proc. 13th Intl. Conf. Pattern Recognition*, vol. 2, pp. 81-85, 1996.

[40] S. Stergiopoulos (Ed.), "Signal Processing Concept Similarities Among Sonar, Radar, and Medical Imaginging Systems – Advanced Signal Processing Handbook," Boca Raton: CRC Press LLC, 2001.

[41] D. Stewart, D. Blacknell, A. Blake, R. Cook and C. Oliver, "Optimal Approach to SAR Image Segmentation and Classification," *IEE Proc. Radar, Sonar Navigation*, vol. 147, no. 3, pp. 134-142, June 2000.

[42] A. Tsai, A. Yezzi, W. Wells, C. Tempany, D. Tucker, A. Fan, W. Grimson, and A. Willsky, "A Shape-Based Approach to Curve Evolution for Segmentation of Medical Imagery," *IEEE Trans. Medical Imaging*, submitted.

[43] C-Y. Tsai, "An Iterative Feature Reduction Algorithm for Probabilistic Neural Networks," *Omega*, vol. 28, pp. 513-524, 2000.

[44] G. Vachtsevanos, W. Daley, B. Heck, A. Yezzi, and Y. Ding, "Fusion of Visible and X-ray Sensing Modalities for the Enhancement of Bone Detection in Poultry Products," *Proc. SPIE Photonics East, Biological Quality and Precision Agriculture II*, Boston MA, Nov. 6-8, 2000.

[45] J. Wang and A. K. Asundi, "A Computer Vision System for Wineglass Defect Inspection via Gabor-Filter-Based Texture Features," *Information Sciences*, vol. 127, pp. 157-171, 2000.

[46] Z. R. Yang and S. Chen, "Robust Maximum Likelihood Training of Heteroscedastic Probabilistic Neural Networks," *Neural Networks*, vol. 11, pp. 739-747, 1998.

[47] A. Yezzi, A. Tsai, and A. Willsky, "Medical Image Segmentation via Coupled Curve Evolution Equations with Global Constraints," *Mathematical Methods in Biomedical Image Analysis, 2000*, *Proc. IEEE Workshop on Biomedical Image Analysis*, pp. 12-19.

[48] Z. Zhang and R. S. Blum, "A Categorization of Multiscale-Decomposition-Based Image Fusion Schemes with a Performance Study for a Digital Camera Application," *Proceedings of the IEEE*, vol. 87, no. 8, pp. 1315 - 1326, Aug. 1999.

[49] S. C. Zhu, T. S. Lee and A. L. Yuille, "Region Competition: Unifying Snakes, Region Growing, and Bayes/MDL for Multiband Image Segmentation," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 18, no. 9, pp. 884-900, Sept. 1996.

VITA

Yuhua Ding was born in Taiyuan, Shan Xi Province, P. R. China in December 1971. She received the Bachelor of Engineering degree in Automation from Shanghai Jiao Tong University in 1994 with a second degree in Applied Mathematics. She joined the Intelligent Control Systems Lab in School of Electrical and Computer Engineering, Georgia Institute of Technology in 1999, where she started to work as a research assistant under the supervision of Dr. G. Vachtsevanos. She received her Master of Science degree in 2001 and Ph.D. degree in 2003 in Electrical and Computer Engineering from Georgia Institute of Technology. Her research interest includes computer vision, snakes, image fusion, multisensory image segmentation, pattern recognition, and artificial intelligence techniques.