uOttawa

L'Université canadienne
Canada's university

FACULTÉ DES ÉTUDES SUPÉRIEURES ET POSTOCTORALES

uOttawa

L'Université canadienne
Canada's university

FACULTY OF GRADUATE AND POSDOCTORAL STUDIES

Ying Huang
AUTEUR DE LA THÈSE / AUTHOR OF THESIS

M.C.S.
GRADE / DEGREE

School of Information Technology and Engineering
FACULTÉ, ÉCOLE, DÉPARTEMENT / FACULTY, SCHOOL, DEPARTMENT

Progressive Image Mosaicking in Wireless Image Sensor Networks

TITRE DE LA THÈSE / TITLE OF THESIS

A. Boukerche
DIRECTEUR (DIRECTRICE) DE LA THÈSE / THESIS SUPERVISOR

CO-DIRECTEUR (CO-DIRECTRICE) DE LA THÈSE / THESIS CO-SUPERVISOR

EXAMINATEURS (EXAMINATRICES) DE LA THÈSE / THESIS EXAMINERS

A. El-Saddik

G. Wainer

W. Gueaieb

Gary W. Slater
Le Doyen de la Faculté des études supérieures et postdoctorales / Dean of the Faculty of Graduate and Postdoctoral Studies

Progressive Image Mosaicking in Wireless Image Sensor Networks

by

Ying Huang

Thesis submitted to the
Faculty of Graduate and Postdoctoral Studies
In partial fulfillment of the requirements
For the M.Sc. degree in
Computer Science

School of Information Technology and Engineering
Faculty of Engineering
University of Ottawa

Your file Votre référence
ISBN: 978-0-494-61347-4
Our file Notre référence
ISBN: 978-0-494-61347-4

Abstract

Prevailing image mosaicking algorithms generate a complete image that incorporates a variety of different images. This variety is captured by different camera sensors. However, these traditional approaches cannot be directly applied to the emerging Wireless Image Sensor Networks (WISNs), because the slow image transmission over a wireless channel causes a noticeable delay before an entire image can be received by its receiver node in a WISN. In this thesis, we propose a Progressive Image Mosaicking Algorithm (PIMA) based on the multi-scan feature of Progressive JPEG (P-JPEG). PIMA's distinguishing characteristic is that it successfully performs mosaicking by using segmental data of images, as opposed to traditional methods, which require the complete data from all images. PIMA mosaics images that are decoded from P-JPEG scans at three levels of quality, and delivers an approximate view of the scene in a short time while the reception of further image data is in progress. Thereafter, it updates the image registration on two other refined levels to gradually enhance the display quality. We developed the concept of Richer Information and Likeliest (RIL) block pair, which is a variation of the Sum of Absolute Difference (SAD) and greatly improves the accuracy of image registration. Experimental results show that PIMA decreases the time delay before the first display of the scene, while preserving an equivalent performance of existing patch-based image mosaicking algorithms.

Acknowledgments

I take great pleasure in thanking many people who have helped me in my studies at University of Ottawa. First, I would like to thank my supervisor, Professor Azzedine Boukerche, for his financial assistance, his invaluable guidance and advice. I would like to thank Professor Abdulmotaleb El Saddik and Professor Gabriel Wainer for their careful reviewing on the thesis and serving on the exam committee, as well as Professor Wail Gueaieb for serving as the chair of the exam committee. I would also like to thank Postdoctoral Fellow Richard Pazzi Werner, and PhD. Candidate Jing Feng, for their kind help in my research work.

Finally, I would like to extend my thanks to my parents, my husband and my friends, whose love and encouragement have always sustained me.

Note it is important to recall that this work and the entire thesis were done under the supervision of Professor Azzedine Boukerche and funded by NSERC Research Fund (PI: Professor A. Boukerche).

List of Publications

The following publications by the author are relevant to the work in this thesis.

1. Progressive image mosaicking in wireless image sensor networks. A. Boukerche, Y. Huang, J. Feng, R. Pazzi. *In Proceedings of the 6th ACM International Symposium on Mobility Management and Wireless Access*, pages 103-110, October 2008.

2. Reconstructing the plenoptic function with wireless multimedia sensor networks. A. Boukerche, J. Feng, R. P. Werner, Y. Du, and Y. Huang. *In Proceedings of the 33rd IEEE Conference on Local Computer Networks (LCN 2008)*, pages 74-81, October 2008.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Image-Based Modeling and Rendering (IBMR) techniques have become widespread for image synthesis in computer graphics because they produce a novel presentation of a composite image. In contrast to their geometry-based counterparts, IBMR techniques render a more detailed view based on rich-content images that represent a real scene. IBMR techniques are used extensively for multimedia applications that involve graphics rendering, such as video games, virtual reality, and visual surveillance. Due to an increasing concern for the security of human lives and property, the demand for real-time remote visual surveillance has grown rapidly. This typical IBMR application retrieves a vast amount of useful information from a sequence of images or video, and offers the user a lifelike view of an image in real time. A network is integrated into a visual surveillance application in order to transmit images or video acquired from a monitored region to a central control-base. Visual surveillance can be applied to a wide range of civilian applications, such as accident detection, traffic control, and crime prevention.

The objective of a visual surveillance application is to provide a user with vivid and continually updated views of a monitored area. However, financial or environmental limitations may render the use of a wired network impossible, thereby, constraining the development of such applications. In addition, because a high volume of required image/video data must be transmitted over a network and processed in order to produce

realistic views, a high computational load is created, and a noticeable time delay may be caused. In such cases, the user has to wait a long time before he can obtain a rendered view of an image. The motivation behind this thesis is to offer a desirable IBMR algorithm able to create a progressive display of an image. The ideal image display will be free from any environmental and perspective constraints, and will have a shorter overall delay as image transmission takes place.

1.1 Motivation

In order for a user to be aware of what is happening in a monitored region, a remote visual surveillance application must offer the user real-time and vivid image views with a magnified FOV of the physical environment. These views are rendered based on images acquired from a far away place and are then transmitted over a network. To design such an application and obtain ideal performance, network deployment, application efficiency, and delay should be carefully considered.

A wired network is the ideal network for a visual surveillance application because it can transmit a large amount of image/video data quickly and reliably. However, cable must first be laid in order to establish the network, and the process of laying cable is costly. It is simply an unrealistic option for low-budget surveillance projects, such as temporary traffic control. Moreover, in certain situations, a wired network is implausible. For example, at the site of a fire, the cable required for data transmission will likely be destroyed. The best way to broaden the scope of compatible environments for a visual surveillance application while also reducing its cost is to use a Wireless Image Sensor Network (WISN) for data transmission. A WISN is a Wireless Sensor Network (WSN) that consists of multiple sensor nodes, each of which is equipped with a camera. A WISN's camera sensors nodes are spatially distributed over a designated area, and are able to capture multiple images of their surrounding physical environment. Since a WISN is wireless, the camera sensor nodes can be placed at any location, and can also

roam to another location without needing a physical cable connection. Thus, the camera sensor node in a WISN can reach a site that wired camera node cannot. For example, a site blocked by an object is inaccessible to a wired network. By deploying a WISN over a monitored region, the visual information in different subareas is easily collected from various viewpoints as the surveillance application captures multiple images. These collected images can be processed by appropriate IBMR techniques, so that a viewer's knowledge of his region of interest can be retrieved. Due to this increased mobility, a WISN can be feasibly used in regions where a wired network is impractical. Flexible deployment of a WISN reduces the cost of network deployment as compared to a wired network, and it also enables temporary surveillance in special environments. As a result, the use of a WISN in a visual surveillance application effectively eliminates the blind-spots that exist in a wired network, remarkably increases the number of applicable environments, and significantly reduces the overall financial cost.

Secondly, the efficiency of a visual surveillance application is correlated with the amount of information contained in each image view; more information indicates higher efficiency. An ideal view for a user's observation is a realistic image display that represents a broad area, which means a view with a magnified Field-Of-View (FOV). However, much like its wired counterpart, the FOV of a single camera attached to a WISN node is a limited one. This small FOV may hinder the deployment of a WISN in certain situations. In order to fully cover the entire monitored region, more camera sensor nodes are required, as is increased financial cost. This unnecessary financial cost can be avoided if camera sensor nodes are optimally distributed using overlapping FOVs. Images captured by each pair of adjacent cameras are then overlapped, and can be stitched together to produce a new and larger image with a broadened FOV. In order to render a realistic view and achieve this enlarged FOV, a suitable IBMR technique is needed for visual surveillance applications. Image morphing and image mosaicking are two IBMR techniques currently under consideration. Image morphing generates an intermediate image based on the similarities of two original images, and without

concern for geometrical errors. The resulting image looks different from either of the two original images [17]. From a monitoring point of view, the ideal virtual environment rendered by the IBMR technique should be an exact replica of the physical environment. Thus, image morphing cannot meet this requirement of surveillance applications. By contrast, image mosaicking renders a larger image in which a number of small images are stitched together, as in mosaics [59]. Since the product of image mosaicking consists of numerous smaller images with limited FOV, this product is a high-resolution image with a broadened FOV. High resolution means that this image contains more information about the monitored area, so that a user can procure more details about the image. The enlarged FOV of the image enables a user to efficiently investigate a wider scope and obtain more information about the monitored area. When compared to the result of image morphing, the product of image mosaicking is comparable. Image mosaicking presents exactly the same scene as the real environment with a larger FOV, which in turn improves the efficiency of a surveillance application. This magnified FOV is achieved by merging several small-FOV images captured by neighbouring cameras, instead of using more camera sensor nodes. Thus, fewer camera sensor nodes are needed, and the financial cost is minimized. In summary, image mosaicking is a desirable method for rendering the virtual environment of a visual surveillance application.

To achieve ideal performance from a visual surveillance application, a WISN is chosen for data transmission from a monitored region to the monitoring centre, and image mosaicking is selected to generate a lifelike view with a enlarged FOV. However, the most critical measurement of the application's performance is the measure of its delay. There is no doubt that the most important requirement for a visual surveillance application is its ability to perform in real time. Nevertheless, since a WISN is used for data transmission, source images are captured from a monitored area and transmitted through low-speed and error-prone wireless links for further processing. The finite bandwidth of wireless links slows down the transmission of images; the unstable wireless connections increase the loss rate of the data package, and the limited energy of a wireless sensor makes it

impossible to resend the images frequently. All of these drawbacks inherent in a WISN culminate in a noticeable delay that occurs just before images are received completely and correctly. In addition, the existing image mosaicking algorithms do not mosaic images until they receive each of the required images completely. Because a great deal of information contained in the multiple images is transmitted through a modem-speed wireless link, the duration of its transmission is much longer. This means that the delay caused by image transmission is increased before an image mosaicking process can start. A user of such a surveillance system has to wait a long time before he can obtain the fully synthesized view. The current approaches, therefore, cannot be directly applied to images received from a WISN. Furthermore, the process of image mosaicking can be computationally demanding and time-consuming. To correctly align all of the images in the collection before stitching them together, the mosaicking algorithm must seek out commonalities between each pair of images. A substantial amount of image information is required in order for these commonalities to be calculated and matched. Therefore, the computational load and time consumption involved are exponentially increased, and this violates the performance requirements of the application.

Reducing the sizes of required images and utilizing a progressive display mode with an early preview are two optimal ways to diminish the delay caused by transmitting images over an error-prone, resource-limited WISN. In order to decrease image sizes, images required for mosaicking must be compressed at the source nodes before they are sent out. Prevalent compression standards can be classified into two categories: *video* and *image*. Video compression standards compress a continuous sequence of still images by only encoding the dissimilarities between two adjacent images, and subsequently producing a motion image. MPEG-2 [45] and MPEG-4 [46] are examples of video compression standards. Even though a movie-like display is ideal for surveillance, the process of a video compression consumes a great deal of a sensor's energy. Furthermore, the simultaneous compression of a group of images is not feasible for transmitting the image group over a WISN. If the data for one image is lost, the next image, even more, the whole image

group cannot be fully decoded until the lost data is retrieved [28]. Data loss happens frequently in an error-prone WISN, meaning that the transmission duration of a group of images is increased. This results in an increased delay before image mosaicking is processed. As a result, video compression standards are impractical for use in a visual surveillance application. Another category of compression standard is the image compression standards. Image compression standards compress still images individually in isolation from all other images, like JPEG [65] and JPEG 2000 [60]. Therefore, even if an image cannot be decoded due to the inevitable data loss, the next image can still be decompressed. In addition, JPEG provides a progressive mode of operation, which works by encoding an image into multiple scans. Each scan contains partial data of a Progressive JPEG image, and is decoded separately. Therefore, each scan of the image can be quickly transmitted, decoded and displayed. Rather than a top-to-bottom display mode, Progressive JPEG offers a coarse-to-fine display mode by decoding and displaying scans one by one. As a result, a rough yet intelligible preview of an image is available much earlier than the complete image. This multi-scan attribute of Progressive JPEG allows a viewer to obtain a basic understanding of an image in a short time based on the image's preview. The image becomes enhanced as the display quality gradually improves. Progressive JPEG outperforms other operation modes of JPEG in a unreliable WISN, because it can provide an early preview based on incomplete image information and gradually improve the quality once more data has been received. The delay that occurs just before the initial display is thus decreased, and the user's waiting time is shortened accordingly. State-of-the-art image transport protocols, such as PCTP [11] and RSTP [6], use Progressive JPEG as their default image format and progressively transmit an image scan by scan. Moreover, the complexity of the image processing is much less than that of the video processing. The computational load and power consumption are not too high for the wireless sensors. In summary, Progressive JPEG provided by the JPEG compression standard is the preferable image format for a real-time visual surveillance application in a WISN.

The aforementioned discussion states that Progressive JPEG is able to shorten the user's waiting time before the first version of an image is displayed, and to provide a coarse-to-fine display mode. However, existing algorithms of image mosaicking do not support Progressive JPEG scans. Even though images are encoded into Progressive JPEG format, current algorithms do not recognize the multiple scans. They consider Progressive JPEG images as those in the default JPEG format, and will not start mosaicking until a complete set of image data is received. As a result, the noticeable delay is not minimized, and Progressive JPEG's desirable coarse-to-fine display is not provided to the user. In other words, the minimum-delay feature that Progressive JPEG offers is not utilized, and thus the user of the surveillance application does not benefit. This provides the motivation for the design of a new image mosaicking algorithm able to support the multiple scans of Progressive JPEG images. The new algorithm should be able to make use of the Progressive JPEG scans in order to eventually decrease the accumulated delay caused by image transmission over a WISN and by the process of image mosaicking. In addition, since each scan contains segmental data of the image, the new algorithm should be able to correctly mosaic images using incomplete information. Moreover, instead of the traditional top-to-bottom display, the proposed algorithm should present the merged image in a coarse-to-fine manner.

In summary, Progressive JPEG is a viable choice because it accelerate image transmission, and offers the user a desirable image presentation mode. If we are to properly enable a real-time visual surveillance application over a WISN that benefits from Progressive JPEG, there is a genuine demand for a new image mosaicking algorithm that supports Progressive JPEG.

1.2 Objectives

The main objective of this thesis is to design and implement an innovative image mosaicking algorithm which correctly mosaics images based on incomplete data, provides the

progressive display, and shortens the delay introduced by image transport over WISNs. To that end, this thesis deploys the following relevant works:

- Inclusive research on the JPEG compression standard, including the Discrete Cosine Transform, the lossless and lossy processes of encoding and decoding, and the operational modes of lossy compression. In addition, the JPEG 2000 compression standard will be studied as well.
- An extensive exploration of techniques for image mosaicking, the state-of-the-art algorithms of image registration will be analyzed.
- The design and implementation of a new image mosaicking algorithm–Progressive Image Mosaicking Algorithm (PIMA). PIMA successfully accomplishes image mosaicking during the reception of the image in order to shorten the delay. Several improvements are devised to advance the accuracy of image registration, which suffers due to incomplete image information.

1.3 Contributions

The contribution of this thesis encompasses the following:

- A profound examination of the compression standards for JPEG and JPEG 2000. The encoding process of Progressive JPEG is stressed as the foundation for the proposed algorithm.
- A comprehensive survey of image mosaicking techniques. Image registration techniques are classified and compared, and the corresponding applicable situations are explored. Approaches that are beneficial to the proposed algorithm are emphasized.
- A novel algorithm, that uses Progressive JPEG as the preferred image format is proposed for the first time as adequate for image mosaicking. By using PIMA,

an image hierarchy is created in accordance with the image quality through the multi-scan feature of Progressive JPEG. The image mosaicking is performed at the coarse level so as to generate a rough view of the merged image in a short time. The accuracy of image registration and the quality of display are refined afterwards. By applying PIMA to a visual surveillance application, delay caused by slow image transmission over a wireless image sensor network is minimized, Field-of-View for observation is enlarged. As a result, the efficiency of a visual surveillance application using a wireless image sensor network is enhanced.

- An innovative similarity metric, Richer information and Likelier (RIL) block pair is developed. RIL block pair is two blocks that not only contain rich prominent information of images, but also best match to each other. By using information of RIL block pair, the accuracy of image registration is improved. The potential misalignment caused by partial image data at coarse level is significantly reduced.

- A desirable display mode for surveillance, coarse-to-fine display mode is introduced. The progressive-quality display allows a user to obtain his basis knowledge of the monitored region at his first glance using a coarse display, and more details when the display quality is refined.

1.4 Thesis Outline

The rest of the thesis is organized as follows:

- In Chapter 2, background information such as the Discrete Cosine Transform, the lossless and lossy encoding and decoding procedures of JPEG and JPEG 2000 image compression standards are introduced. At the same time, Progressive JPEG is explored in depth.

- Prevailing algorithms, including patch-based and feature-based approaches to image registration for image mosaicking, are studied in Chapter 3. The most common

measures used for image registration are also investigated.

- In Chapter 4, the design and the framework of the proposed algorithm, PIMA, are presented. Four distinguishing features, which are Scan Preference, Image Blocking Search (IBS), Richer Information and Likelier (RIL) block pair, and Small Range Search (SRS), are described as well.

- The workflow and the implementation of PIMA are discussed in detail in Chapter 5 according to the aforementioned four features. Problems addressed during the implementation are also mentioned.

- Chapter 6 explores the design of experiments, analyses results, and compares performance between PIMA and some current mosaicking algorithms.

- Lastly, Chapter 7 concludes the thesis and addresses some potential future work related to the developing applications.

Chapter 2

Image Compression Standards

The slow image transmission over a WISN in a visual surveillance application results in a noticeable delay before a realistic view is displayed. The size of a digital image file is greater than the size of a text file because a vast amount of data is needed to represent the image. The image file not only requires a large storage space, but also slows down the transmission of the image over a WISN. In order to increase the transmission speed and shorten the delay, the digital image is always compressed into a smaller size before it is sent. Currently, the most prevailing compression standard for images is JPEG, which is supported by most image processing applications. JPEG 2000 is another popular compression standard, with more innovative functions allowing images to be processed more conveniently and effectively.

2.1 The JPEG Compression Standard

JPEG is an acronym for the Joint Photographic Experts Group, the organization that issued a compression standard for digital still images in 1992 [65]. This image compression standard was named JPEG, and was approved as an international standard by the International Organization for Standardization (ISO) in 1994. The JPEG compression standard became prevalent because it effectively reduces the size of an image file

while preserving almost equivalent quality. Most applications compress images using the JPEG standard in order to reduce the image's memory consumption or to accelerate image transmission over a network.

The purpose of JPEG encoding is to compress an image in another image format, such as a bitmap, into JPEG format; the purpose of the JPEG decoding is to decompress an JPEG image into another image format. The JPEG compression standard defines two types of encoding and decoding processes: lossy and lossless coding processes. The lossy coding process uses the Discrete Cosine Transform (DCT) in order to achieve high-ratio compression without significantly downgrading image quality. Therefore, lossy coding is also known as a DCT-based process. On the other hand, the lossless coding process is not DCT-based, and is therefore used by applications where lossless image compression is required.

2.1.1 Lossless Coding

Just as its name implies, lossless coding compresses an image without discarding any image data. The decompressed image is the exact copy of the original one. Instead of using DCT, the lossless coding predicts the value of a pixel using a predictor combined with the values of up to three neighbour pixels. The difference between this prediction value and the actual value for each pixel is then encoded through entropy coding. The lossless decoding process decompresses a JPEG file in the reverse way of the encoding process. Figure 2.1 illustrates this lossless encoding process of JPEG.

Entropy coding saves the image data using a compacted bit string in order to achieve data compression. It uses the shortest codewords to present the most common symbols, and the longer codewords to decipher the less common symbols. The use of entropy coding is irrelevant to the media's attributes or the semantic of input data. Entropy coding is a lossless approach because it only changes the input data's presentation using a shorter form, and no data is discarded during the process of entropy coding. By applying the entropy coding, the entire image data is converted and stored in a compressed form.

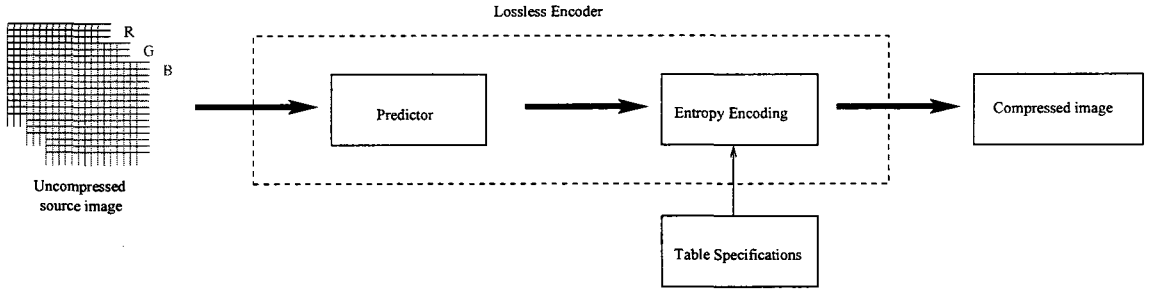


Figure 2.1: The lossless encoding process of JPEG

Lossless coding is an operation mode that is independent of DCT. It compresses an image only using entropy coding, so that all the information carried by the encoded image is the same as the information of the original one. Clearly, lossless coding is the method with the lower compression ratio when compared to the lossy coding of JPEG compression standard.

2.1.2 Lossy Coding

In contrast to lossless coding, lossy coding obtains a high compression ratio by discarding the information that is insignificant to the quality of an image. As a result, the image is compressed into a fairly small size without visibly degrading quality. Although there are a number of methods for encoding an image into a JPEG file at a high compression ratio, a DCT-based encoding method is the most conventional way for applications to produce the best quality picture.

2.1.2.1 The Discrete Cosine Transform (DCT)

As with other transforms used in image processing, the purpose of the Discrete Cosine Transform (DCT) [34] is to remove the correlations among the image data. After decorrelation, each coefficient produced by the transform can be encoded without any dependence on others, and the efficiency of the compression is not reduced. DCT type II (DCT II) is a two-dimensional space version extended from the original DCT, and is the

most common variant used for image compression, as in the JPEG standard. It has two versions: Forward DCT (FDCT) and Inverse DCT (IDCT), which are used for JPEG encoding and decoding processes.

A valuable property of DCT II for image compression is *Energy-compaction.* This property enables the correlated image data to be compacted into a low-frequency region, and the uncorrelated image data to be presented in the high-frequency area. The JPEG compression standard employs DCT in order to transform the most information of an image into the DCT coefficients, and to pack them into a low-frequency area. The high-frequency DCT coefficients are then discarded by the quantizer because they are not vital information for image quality. Because some image information is lost during the compression process, the image is compressed at a high compression ratio while the quality is preserved. Thus, the DCT-based coding process is a lossy process, and it dominates most of the JPEG compression standard.

2.1.2.2 The Process of DCT-based Encoding

The DCT-based encoding process is composed of *Colour Space Transformation*, *Down-sampling*, *Discrete Cosine Transform*, *Quantization and Entropy Coding.*

Colour Space Transformation: In general, sensor cameras capture colour images in the RGB colour space. Three channels of colours: red, green, and blue, are used for the image presentation in the RGB colour space. Each pixel in the image is presented by the combination of different values of R, G and B. These RGB images are first transformed into images in the YCbCr (or YUV) colour space. YCbCr has three different components: Y represents luminance while Cb and Cr refer to chrominance. For each pixel in the image, the value of Y is computed by adding designated weights to values of R, G and B. The value of Cb is produced by subtracting the value of Y from the value of B, and then scaling. The Cr value is obtained from the R value in the same way using different scaling factors. By applying colour space transformation, brightness

information is separated from an image's colour information in the YCbCr colour space, so that a high-ratio compression of chrominance components can be performed.

Downsampling: Research shows that human eyes are sensitive to brightness. Human beings can see more variation in brightness than they can in colour. In other words, our fault tolerance for chrominance is greater than that of luminance. Based on this knowledge, the chroma components Cb and Cr can be minimized at horizontal and vertical downsampling rates so as to achieve great compression without affecting our perception of this change. Some colour information might be lost in downsampling, yet it is invisible to us in the decoded image. Different downsampling ratios enable space-saving at percentages ranging from 33% to 50% [61]. The rest of the encoding processes of the three components (Y, Cb and Cr) occur separately, and are carried out in a similar manner to the following three steps.

Forward DCT: After chroma components are downsampled, each of the three components is then split into 8×8 pixel blocks. The Forward Discrete Cosine Transform (FDCT) is then employed to convert the data of each block from spatial domain to frequency domain [49]. After FDCT is performed, the most information of a pixel block will be concentrated to a few elements of DCT output. FDCT generates 8×8 DCT coefficients, each of which represents a portion of the image. The first element of a DCT block is named the Direct Current (DC) coefficient, which refers to the zero frequency; the following 63 coefficients are named Alternating Current (AC) coefficients, referring to the higher frequencies. AC coefficients are the deviations of the DC coefficient. By applying FDCT, most of the low-frequency coefficients are packed in the top-left area of a DCT block. Small, high-frequency elements can thus be discarded for compression.

Quantization: The purpose of quantization is to achieve greater compression by discarding information that is insignificant to the image quality. Human eyes can easily

notice slight changes of brightness within a large region, but are less able to recognize the difference of the strength of high-frequency brightness variations. Thus, a great amount of information in the high-frequency coefficients can be eliminated. Each DCT coefficient is divided by the corresponding quantizer step-size and rounded to the nearest integer in the quantization step. Consequently, most high-frequency coefficients are rounded to zero, and the remains are changed into small positive or negative integers. The number of bits required for storing these quantized coefficients is then reduced, and the amount of data needed to represent the image is significantly decreased. The quantizer step-sizes are stored as constants in a table, which is a quantization matrix. Table ?? [61] shows a common quantization matrix. In order to save the storage space and maintain the image quality at a satisfactory level, the quantization matrices are particularly defined to preserve an image's low-frequency information while discarding its high-frequency information. In addition, the quantization matrices can be customized to meet the needs of applications. A heavy compression matrix blurs most of the details by converting more high-frequency coefficients to zero. A high compression ratio is then achieved while much detailed information is lost. Since some information about the details is discarded in order to greatly decrease an image's storage space, quantization is the primary lossy procedure during the entire DCT-based encoding process.

Entropy Coding: Like the lossless encoding process, the purpose of the entropy coding is to present the image data in a compacted form. Entropy coding has two steps: *Run Length Encoding (RLE)* and *Huffman Coding.* After quantization, quantized DCT coefficients are reordered in a one-dimensional array by traversing the original array in a zig-zag sequence. After this rearrangement, RLE replaces the same consecutive data by a single data and the number of its occurrence. The original array of the quantized DCT coefficients is then represented in a shorter way. Lastly, Huffman coding is applied and results in further compression for the image data. Huffman coding deciphers the most common elements using shorter bit strings based on the probability of its occurrence

in the RLE array. Through applying RLE and Huffman coding, the two-dimensional quantized DCT coefficients are converted into a much shorter bit sequence, and the required storage space for these coefficients is reduced to a minimal level. Therefore, high compression ratio is achieved.

These five steps involved in compressing image data are in accordance with Baseline JPEG encoding, which is the default operation mode of JPEG. Figure 2.2 illustrates the compression process. A high compression percentage is gradually accomplished, allowing for image quality preservation.

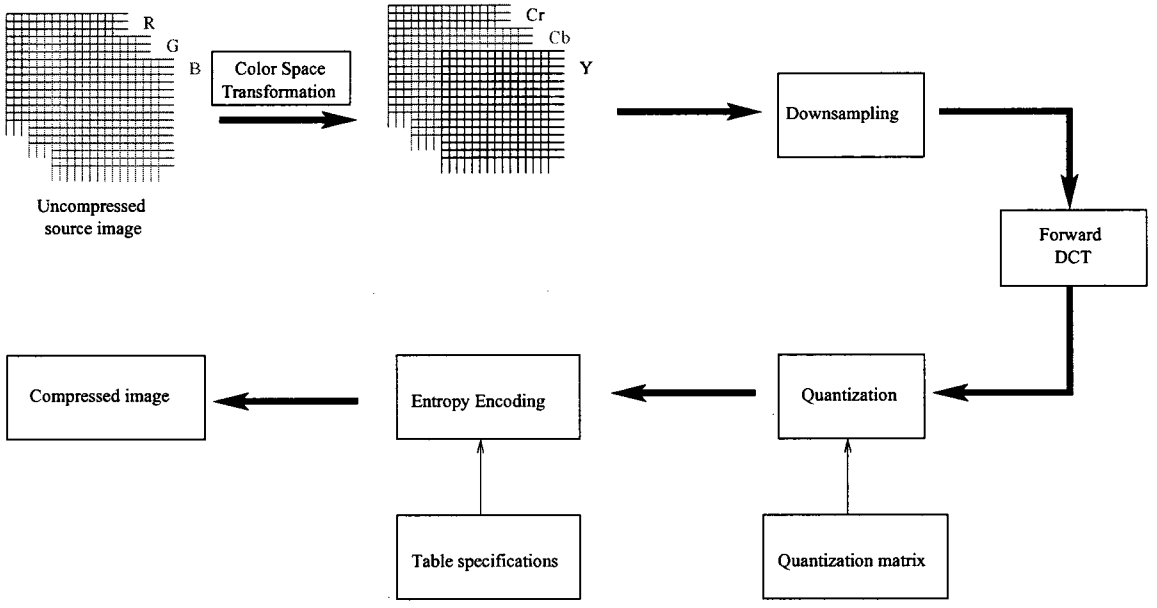


Figure 2.2: The DCT-based encoding process of JPEG

2.1.2.3 The Process of DCT-based Decoding

Similar to the encoding process, the process of DCT-based decoding also contains five steps: *Entropy coding, Dequantization, Inverse DCT, Upsampling, and Colour Space Transformation.* The decoding process is exactly the reverse process of the DCT-based encoding and will be discussed in brief. The process of decoding a JPEG image begins with entropy decoding, which uses Huffman decoding and Run Length Decoding to

retrieve quantized DCT coefficients from the bit sequence and then save them into a one-dimensional array. The quantized DCT coefficients are then rearranged into 8×8 blocks in a zig-zag route. In the dequantization step, the quantized coefficients in the 8×8 block are dequantized using the same quantization matrix specified in the encoding process. The DCT coefficients are then retrieved. After the dequantization, Inverse DCT (IDCT) is applied to convert the DCT coefficients in frequency domain into the image data in spatial domain. Thereafter, upsampling of chroma components resizes the components Cr and Cb to their original size. Finally, the colour space of the image is transformed from YCbCr into RGB. As a result, a bitmap version of the JPEG image is reconstructed. However, since some information is discarded in the quantization step of the encoding process, this information cannot be retrieved. The recovered DCT blocks are, therefore, slightly different from the original blocks; the reconstructed bitmap image is not exactly the same as the image before being compression. Still, the difference between the original image and the decompressed one is invisible to the human eye. Figure 2.3 exhibits the decoding process.

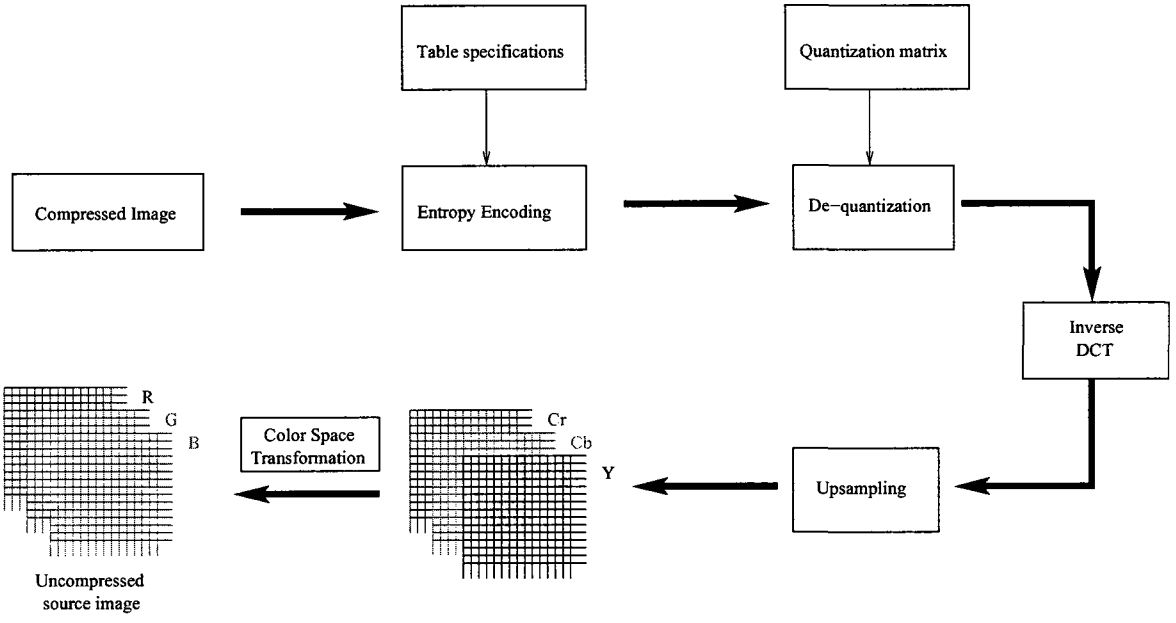


Figure 2.3: The DCT-based decoding process of JPEG

2.1.2.4 Operation Modes of Lossy Encoding

In order to be compatible with different kinds of digital images and to work with a wide range of applications, JPEG offers three possible modes of operation for image lossy encoding, so as to satisfy the requirements of image processing applications.

- **Baseline Encoding (Baseline JPEG)** – The image file is encoded into a single scan where data is stored top-to-bottom and left-to-right. The codec corresponding to the baseline encoding and decoding processes provides a sophisticated compression approach that is effective and sufficient for most image processing situations. Thus, baseline encoding is the default mode of operation, and is supported by the majority of applications.

- **Progressive Encoding (Progressive JPEG)** – The image data is divided and encoded into a series of scans which are transmitted sequentially. An initial view is generated when the first scan of an image is decoded. The subsequent scans add more image data to the current scan in order to increasingly refine image quality. The image is displayed in a coarse-to-fine manner.

- **Hierarchical Encoding (Hierarchical JPEG)** – Multiple-resolution versions can be generated by hierarchically encoding the image. Therefore, lower-resolution images can be first used before the full resolution version is ready for application.

2.1.2.5 Progressive JPEG

As stated in section 2.1.2.4, Baseline JPEG is the most widespread JPEG operation mode and image format, and it stores image data in a top-to-bottom, left-to-right scan. In contrast, Progressive JPEG uses progressive encoding as its mode of operation. It encodes an image into a series of scans and transmits them sequentially, providing a gradual coarse-to-fine view of the image.

Generally, the size of an image used in multimedia applications is large as compared to a text file. Thus, the transmission of a whole image is time-consuming, and takes

even more time in low-speed WISNs. Progressive JPEG becomes attractive because it divides the whole image file into multiple scans, and the size of each scan is smaller than the size of the whole file. Therefore, the earlier scans can be transmitted and displayed quickly, while the latter scans are still being received. This multi-scan attribute enables application users to have a rough image preview in a short time. When more data is received from the latter scans, users can obtain increasingly clearer views after waiting longer. This progressive display mode of Progressive JPEG is also much better than the top-to-bottom display mode of Baseline JPEG, because the user can decipher the content of an image quickly. In addition, each scan of a Progressive JPEG file can be decoded separately into an image, and the quality of all decoded images increases, as Figure 2.4 illustrates. The multi-scan attribute of Progressive JPEG allows applications to use the multiple scans for advanced image processing, as in image mosaicking. Furthermore, when waiting for refined views of the current image, users have the option of switching to the next image if the current view is acceptable to them. Since the approximate preview is provided quickly, the user's waiting period is then shortened significantly. Progressive JPEG is not explicitly helpful compared to Baseline JPEG if images are transmitted over high-speed network connections. Notwithstanding this, most surveillance applications over WISNs exchange image data at a slow, modem-speed rate. Clearly, Progressive JPEG is more desirable and feasible for surveillance purposes than Baseline JPEG.

Progressive JPEG encoding has the identical first four steps as the Baseline JPEG process: *Colour Space Transformation, Downsampling, DCT* and *Quantization.* After quantization, Progressive JPEG introduces an image-sized buffer to temporarily save quantized DCT coefficients. However, the entropy encoding differs from the Baseline JPEG process. Instead of encoding all coefficients of a DCT block in a zig-zag order into a single scan, the buffered coefficients are divided into several groups, each of which is encoded into a scan. Spectral selection and successive approximation [65], as well as a combination of the two are common approaches used to partition quantized DCT coefficients and to form multiple scans of the image with increasing quality levels.

- **Spectral Selection**: 64 DCT coefficients in each 8×8 DCT block are divided into a set of particular spectral bands. The same-spectrum bands of all DCT blocks are grouped into one scan and then sent to the encoder. For instance, the DC coefficients of each DCT block are grouped into the first scan and transmitted to the encoder, followed by the first nine AC coefficients from the second scan, and so forth. Figure 2.5 shows an instance of multiple scans of a DCT block using spectral selection. These scans are transmitted to the receiver sequentially. Lower-frequency bands in the first scan are received and decoded into an approximate view of the image, at the same time, the later higher-frequency scans are being transmitted. The display quality of the image is improved as each incoming scan is received and decoded.

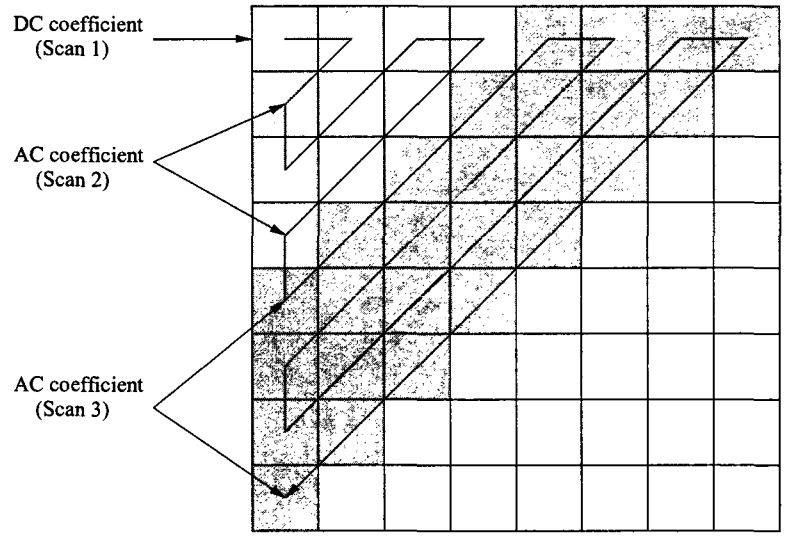


Figure 2.5: Multiple scans formed by spectral selection

- **Successive approximation:** Rather than processing the sequence of complete coefficients one by one, successive approximation forms multiple scans by using different bits of 64 DCT coefficients. The process of some bits of a coefficient is postponed initially and retrieved in later scans. For example, a specified number of the most vital bits of all coefficients are encoded first, and the less significant

bits are then passed to the encoder in subsequent scans. Figure 2.6 [65] shows an example. Moreover, spectral selection and successive approximation can be employed either separately, or in tandem.

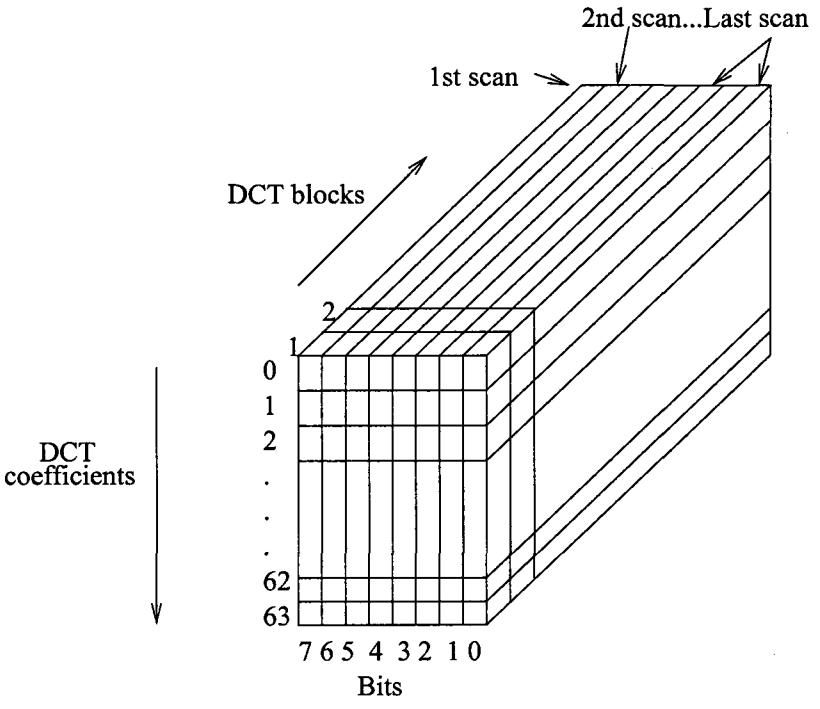


Figure 2.6: Multiple scans formed by successive approximation

In summary, the whole file is divided into a number of scans in Progressive JPEG by using spectral selection, successive approximation or a combination of the two. Each scan is smaller in size than the whole file, and can therefore be transmitted faster over networks. A low-quality but recognizable version of the image can be exhibited very quickly, and gradual quality refinements follow. This coarse-to-clear presentation mode is more desirable than Baseline JPEG's slow, top-to-bottom mode. The Progressive JPEG image is therefore the better choice used for visual surveillance applications that transmit images over a error-prone and bandwidth-limited WISN.

2.2 The JPEG 2000 Compression Standard

After the JPEG compression standard was issued, the Joint Photographic Experts Group committee released a new wavelet-based image compression standard in the year 2000: JPEG 2000. In contrast to the DCT used in JPEG, JPEG 2000 [51] is grounded in the Discrete Wavelet Transform (DWT). The DWT represents the signal's most salient information in high amplitude and its less significant information in low amplitude. The low amplitudes are eliminated when data is compressed. Wavelet compression is good at representing the momentary components of a data signal using a smaller amount of information. Thus, compared to JPEG, JPEG 2000 is able to occupy as little storage space as possible.

The compression performance of JPEG 2000 exceeds the performance of JPEG in negligible artifacts with almost no visible blocking. The image decoded by JPEG 2000 looks more natural than the one decoded by JPEG when the compression ratio is higher. In addition, JPEG 2000 allows the extraction of multiple scalings or components from the same compressed codestream. This is attributed to the employment of the DWT and a more complicated entropy encoding algorithm, including scalar quantization, context modeling, arithmetic coding and post-compression rate allocation. This particular attribute results in a number of specific advantageous features [60, 51] such as multiple resolution representation, progressive transmission, lossless and lossy compression, random codestream access, and processing and error resilience.

Though JPEG 2000 has developed better compression performance and a number of new functions, noticeably high computational overhead and memory demands are the obstacles in its popularization for applications.

2.3 Summary

For the sake of accelerating the image transmission over a WISN and shorten the delay in a visual surveillance application, two image compression standards have been intro-

duced in this chapter. The coding processes of the JPEG compression standard has been classified into two types: lossless and lossy. The process of lossy compression has been examined closely because it constitutes most of the JPEG standard. The Discrete Cosine Transform has been introduced because it is the basis of lossy coding. Thereafter, the DCT-based encoding process has been presented in detail, and the DCT-based decoding process has been briefly discussed because it is the reverse approach of the encoding process. Three types of operational modes for lossy encoding and their associated attributes have also been presented. After that, a further study of progressive encoding has then been conducted, and features deemed to be beneficial for image transmission has been exposed. As a result, Progressive JPEG proves to be adequate for image transmission over a latency-sensitive and bandwidth-limited WISN, and provides a desirable coarse-to-fine display mode for a visual surveillance application. Lastly, the JPEG 2000 standard has been explored as well. While JPEG 2000 has not been broadly used in current Web browsers thus far, it could be considered in the future stages of the proposed algorithm.

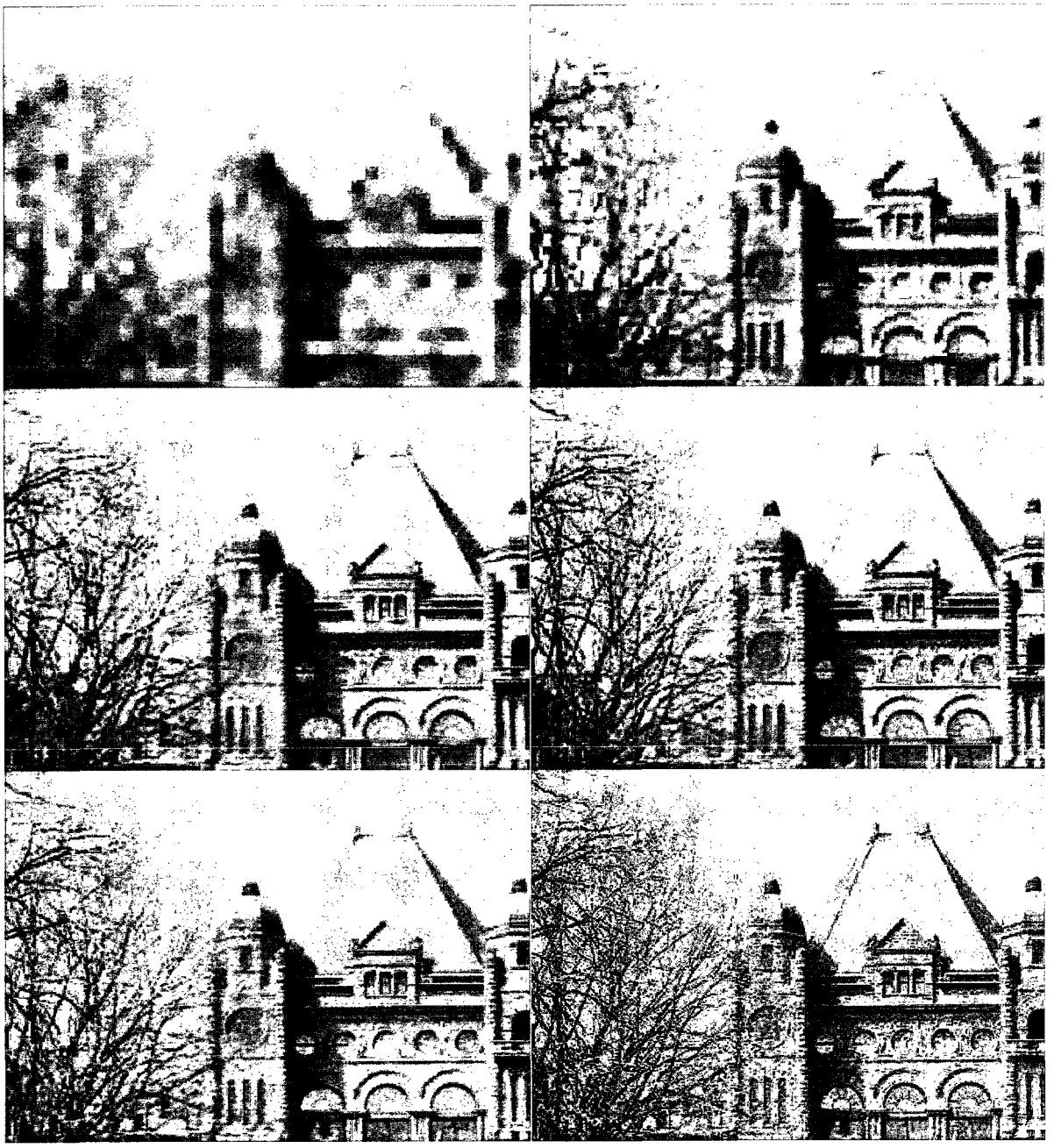


Figure 2.4: Coarse-to-fine views decoded by Progressive JPEG

Chapter 3

Survey of Image Mosaicking Techniques

Image mosaicking in the field of image processing is derived from photographic imaging. Photographic imaging partitions a large photo into a set of rectangular cells, each of which is an individual picture. The opposite process, image mosaicking, merges a number of overlapping pictures to compose one large photograph. This process is used extensively for the construction of a high-resolution and panoramic image. A visual surveillance application uses image mosaicking in order to achieve a broad Field-Of-View (FOV), so that a user can efficiently investigate a monitored region. In this chapter, we briefly review current image mosaicking techniques, and categorized them according to steps of the general mosaicking process.

3.1 Introduction to Image Mosaicking

In order to render a large view with a magnified FOV, a visual surveillance application acquires a series of overlapping images from a real scene, and transmit them to work-stations in the monitoring centre. Once the images are collected, they may be rotated, shifted or distorted since they are captured by different cameras located in different posi-

tions. *Motion modeling* is the process of building the mathematical relationship between two adjacent images' coordinates. These relationships allow for all pairs of images to be matched and aligned appropriately. The step of image matching and alignment is called *Image registration.* During image registration, each pair of images in the collection is examined so as to identify the degree of similarity. The common feature of both images or the likeliest block-pair between two images is sought based on a proper similarity metric. The common information exposes the overlapping status of the two images. Image registration repeatedly seeks the matching segments in each pair of images, and aligns these images according to the information held by the common segments. *Image composition* is the final step, in which all of the images are stitched together according to the information collected from image registration. Figure 3.1 presents the procedure of image mosaicking. Figure 3.2 illustrates an entire process of a visual surveillance application using traditional image mosaicking in a WISN. State 1 in Figure 3.2 shows the images captured by camera sensors are encoded into single-scan Baseline JPEG (B-JPEG) versions. In State 2, 3, and 4, Each of these B-JPEG images is read from top to bottom and saved as a bit-stream, and is transmitted to the workstation in the control base. Traditional mosaicking approaches will not start mosaicking until all of these images are received completely. State 5, 6 and 7 presents traditional methods display a wide-angle image from top to bottom after they completely obtain all of the images from different camera sensors. Discussion in Section 1.1 states that traditional methods are unsuitable for image mosaicking in a WISN because of the potential delay caused by image transmission over unreliable wireless connections. In order to adapt current mosaicking approaches to a surveillance application in a WISN, up-to-date techniques will be surveyed in the following sections in accordance with the steps of image mosaicking.

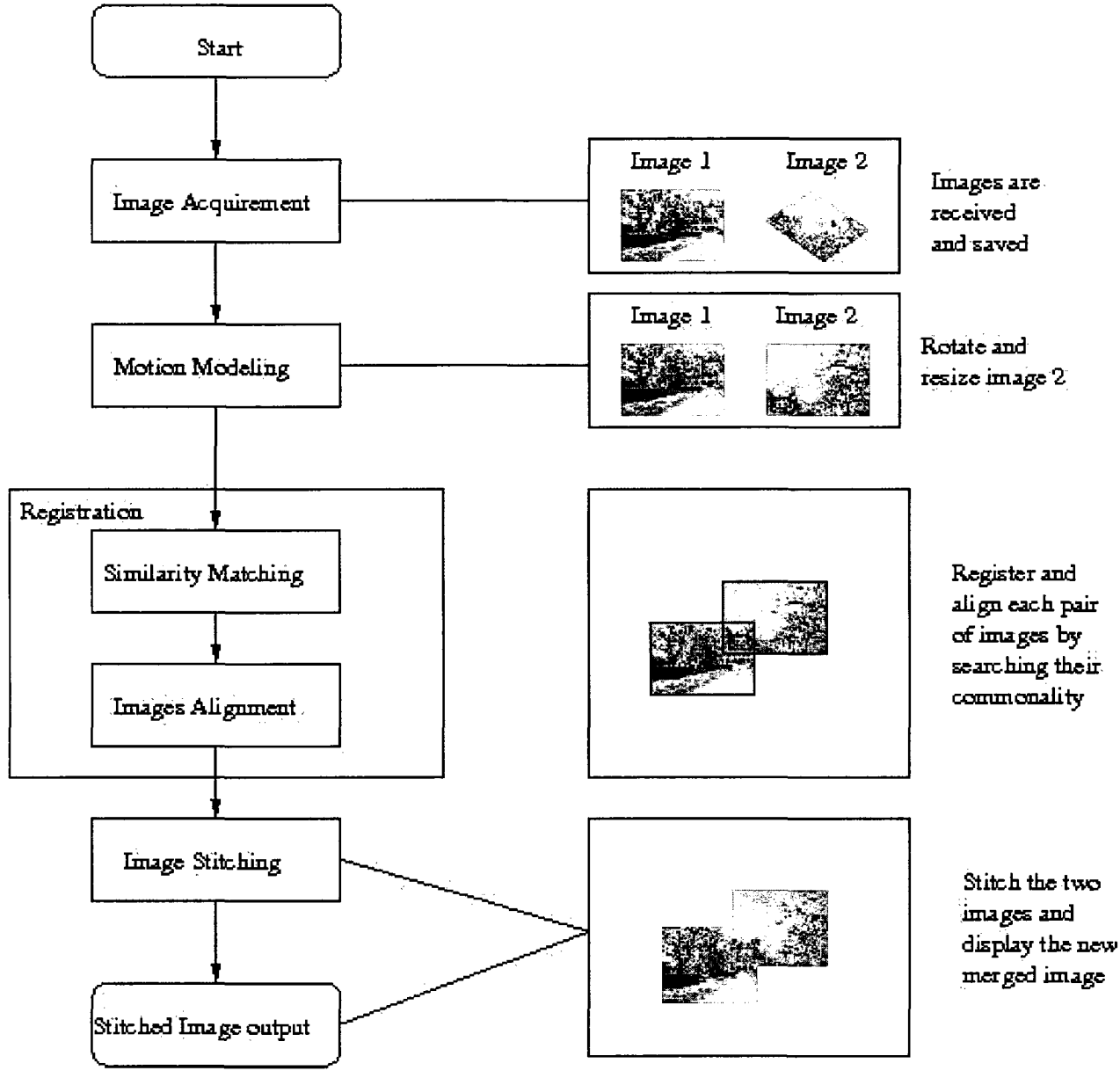


Figure 3.1: The flowchart of image mosaicking

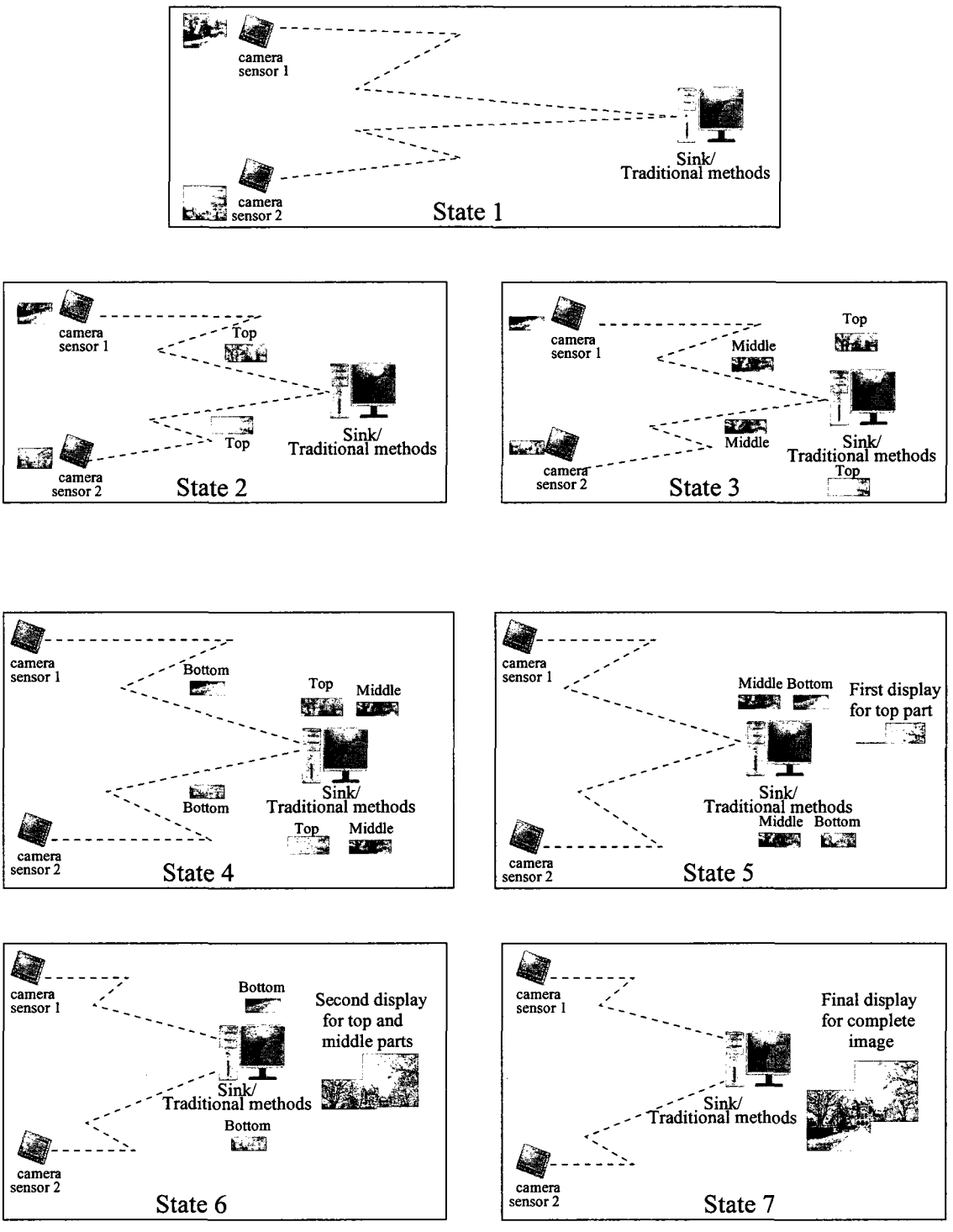


Figure 3.2: An entire process of traditional image mosaicking in a WISN

3.2 Motion Modeling

Camera sensors operate in various locations and orientations, and may be panning or zooming when taking pictures. Therefore, different geometrical transformations are found in images used for mosaicking. For the sake of locating a common object in two overlapping images, the relationship between the pixel coordinates of the two images is needed. This is named motion modeling. A variety of parametric models representing different motion transformations are required to create the mathematical relationship. Motion transformations can either be categorized into 2D and 3D transformation or Cylindrical and Spherical Coordinates as an alternative. Image distortion can simultaneously be corrected at this step.

2D Transformation, also known as planar transformation, is the conversion that occurs on a 2D plane. It consists of translation, affine and projective or any combination of two or three. Figure 3.3 illustrates some 2D transformations of a square. Hartley and Zisserman [27] have clearly described the coordinate conversion according to each of the transformations. The coordinate conversions use 3×3 conversion matrices multiplied by homogeneous or projective coordinates, which are generally used to represent pixels on a 2D plane. *3D Transformation*, has the similar converted components as 2D transformation, but the degree of freedom of each 3D transformation differs from the 2D one. Okutomi et al. [47] used a 4×4 projection matrix for coordinate conversion without discarding the z-axis values in 3D cases. Foley et al. [19] used an orthonormal rotation matrix and a 3D translation vector to project pixels in 3D transformation onto a 2D plane with a certain distance along the z-axis. This allows the coordinate conversion to be done even if camera parameters such as calibration and/or orientation parameters were unknown.

Cylindrical and Spherical coordinates are optional coordinate systems for establishing mathematical relationship instead of homogeneous or projective coordinates of 2D/3D motions. As Szeliski described in [59], the collected images can be projected onto the

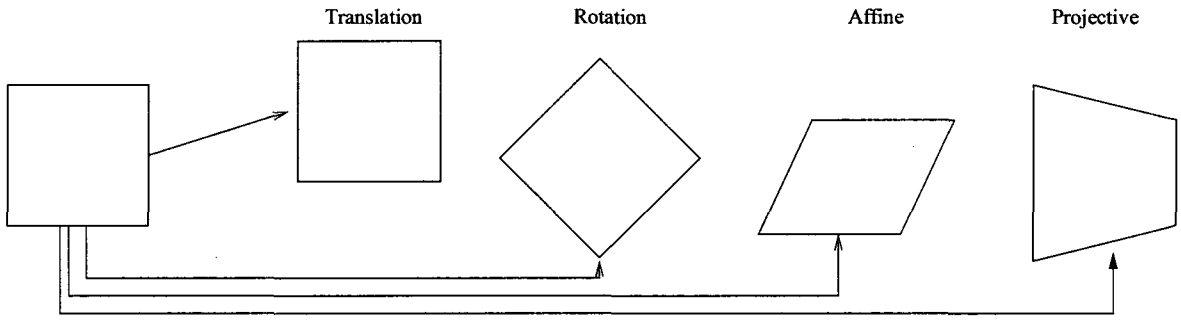


Figure 3.3: 2D transformations of translation, rotation, affine and projective for a square

same cylindrical or spherical surface, and pixels in the two images are represented using the same coordinates. Cylindrical warping is preferable when images have undergone a minor horizontal shift. Spherical projection, however, is desirable in cases where spherical or hemispherical views are parts of the panoramic output. The computational overhead is reduced by using cylindrical or spherical coordinates in such specific situations.

Image distortion happens when an image is captured by a camera with a wide-angle lens. In most cases, the distortion is radially symmetric. and is known as radial distortion. Barrel distortion and pincushion distortion are two types of radial distortion. To estimate the specific degree of radial distortion, a plumb line method [33, 40] was first proposed. This method uses distortion parameters to indicate the degree of distortion, and modifies them to straighten all the distorted lines. The correction of the distortion also can be accomplished during the process of image alignment. Sawhney et al. [52] applied a quadratic distortion correction term when images were aligned by means of intensity-based minimization. A feature-based method coupled with the quadratic radial distortion used by Stein [56] is more suitable for general situations, but its complexity results in greater computational load.

Motion modeling is responsible for coordinate system conversion depending on the 2D/3D transformations among the collected images. In some situations, low complexity cylindrical or spherical projections can produce results of equal quality to the 2D/3D transformations. Image distortion can also be corrected in this step. After motion

modeling, geometrical errors caused by the motion transformation of each pair of images are appropriately corrected by mapping the pixel coordinates from one image to another.

3.3 Image Registration

Image registration, also known as image alignment, is the process of geometrically aligning images captured from different viewpoints or by different camera sensors in a monitored region. It compares any two images in the collection that have a similar characteristic. The location of the commonalities indicates the overlapping status of the two images. By examining every pair of images in the collection, all of the images can be aligned properly according to the positions of their similarities. Image registration is a crucial process in image mosaicking because it provides the basis for stitching images together in the appropriate geometrical order. A large number of pixels are involved in the comparisons and calculations needed to find the common traits. Thus, image registration generates great computational overhead during image mosaicking.

Due to its significance in a variety of image mosaicking applications, image registration has become a popular issue in recent research. Many advanced algorithms have been developed to improve either the accuracy or efficiency of image registration. Existing approaches can be classified in accordance with different criteria, for example, the application's purpose or computational cost. In this thesis, algorithms are categorized into two groups: *Patch-based* and *Feature-based*, according to the rationales of image registration algorithms. Patch-based algorithms search common fractions on a patch basis, and only the degree of the difference in intensity is taken into account. Feature-based algorithms consider the common fraction to be the same feature. This feature has distinguishing information such as the orientation or the curvature of a specific object.

3.3.1 Patch-based Registration

Patch-based algorithms, or pixel-based algorithms, define a pixel block as being a certain size, and compare the pixel intensity-difference of each possible block-pair in two images. A similarity criterion is used to measure how similar the pixel blocks are to each other. The most alike block-pair contains the most similar characteristics of both images. The two images can then be aligned and merged properly according to the positions of the two blocks. A suitable similarity metric and a robust search technique are the key for patch-based algorithms. Various approaches and improvements of patch-based registration have been devised.

3.3.1.1 Similarity Metrics

Similarity metrics are responsible for assessing intensity differences between all pixel blocks in every pair of images. A great deal of computation is generated because the calculation happens pixel by pixel. An ideal similarity metric should be able to precisely represent the similarity-degree of each block-pair using a low computational cost. Sum of Squared Difference (SSD), Sum of Absolute Difference (SAD) and Normalized Cross-Correlation (NCC) are the prevailing criteria for successful matching of each block-pair.

Sum of Squared Difference (SSD) SSD, also known as the sum of squared deviations, is widely used to compute the deviation between the arbitrary datum and the mean of a set of data in statistics. From an image processing point of view, SSD is applied to estimate the difference between a pair of pixel blocks in two images. Each pixel in each block is compared on the basis of pixel intensity. Given an image $k+1$ and its reference image k, and that the size of the pre-defined block is $m \times n$, $I(i,j)$ is the intensity of a pixel at position (i,j). Equation 3.1 shows its SSD function.

$$SSD_k(i,j) = \sum_{i=1}^{m}\sum_{j=1}^{n}(I_{k+1}(i,j) - I_k(i,j))^2 \tag{3.1}$$

The block-pair with the smallest SSD value will have maximum similarity. This means that the block-twin contains the most common traits. The location of the blocks in both images is used for image alignment. To discard pixels that are useless for the similarity estimation, a spatially per-pixel weight [59] is used to form a weighted SSD function, so that pixels are weighted according to their contributions to the image alignment. Several robust or efficient functions proposed by Stewart [57] can be used to replace the squared term in order to make the above metric more efficient. Sum of Absolute Difference is one such function.

Sum of Absolute Difference (SAD) Since the computational overhead and memory consumption of multiplication are considerably great, SAD [57] was devised to reduce the complexity of SSD function. As equation 3.2 shows, the squared function is replaced by the absolute function, making the equation more effective. Thus, the computational load is remarkably reduced. Like SSD, the block-pair with the minimum SAD means that the blocks have significant commonalities.

$$SAD_k(i,j) = \sum_{i=1}^{m}\sum_{j=1}^{n} |(I_{k+1}(i,j) - I_k(i,j))| \tag{3.2}$$

SAD is an exceedingly fast metric as compared to other similarity metrices. The absolute function that SAD uses is simple, and generates minimum computational overhead even though SAD takes into account each pixel in a block. Thus, SAD is an efficient metric for a broad search or comparison of a large number of blocks. For example, SAD is the suitable similarity metric when an exhaustive search is needed in order to search the likeliest block-pair for two overlapping images. The exhaustive search is accelerated because the computational load produced by SAD is negligible. Due to its simplicity and efficiency, SAD is a proper similarity metric for image registration so as to improve the performance of a surveillance application in a wireless sensor network.

Normalized Cross-Correlation (NCC) NCC is derived from Cross-Correlation (CC), which is simplified by the expansion of SSD function. Unlike with SSD and SAD,

the most similar block-pair is the pair with the greatest CC value. However, CC cannot avoid intensity-variation errors caused by exposure differences when bright patches exist in either image. NCC, as equation 3.3 shows, uses the mean values of pixel blocks to overcome the aforementioned imperfection of CC.

$$NCC_k(i,j) = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n}((I_k(i,j) - \overline{I_k}(i,j))(I_{k+1}(i,j) - \overline{I_{k+1}}(i,j)))}{\sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}((I_k(i,j) - \overline{I_k}(i,j))^2(I_{k+1}(i,j) - \overline{I_{k+1}}(i,j))^2)}} \tag{3.3}$$

where

$$\overline{I_k}(i,j) = \frac{1}{mn}\sum_{i=1}^{m}\sum_{j=1}^{n} I_k(i,j) \tag{3.4}$$

and

$$\overline{I_{k+1}}(i,j) = \frac{1}{mn}\sum_{i=1}^{m}\sum_{j=1}^{n} I_{k+1}(i,j) \tag{3.5}$$

In general, the value of NCC is constrained to fall in the range of $[-1, 1]$, which makes it suitable for similarity measurement in some advanced applications. Nevertheless, the NCC function has a higher complexity than the functions of SSD and SAD.

A suitable and carefully-chosen metric is the essence for correctly evaluating the similarity of two images. A number of improvements have been declared to make these criteria more efficient [50, 29, 25, 5], and to correct the exposure difference between pixels [21, 22, 39, 2]. In the case of a colour image, the intensity difference for each of the three colour channels is estimated separately by the selected criterion, and summarized for similarity assessment.

3.3.1.2 Image Alignment Approaches

After the similarity metric is confirmed, a block-matching search using a minimum computational cost has to be determined. The most intuitive method is an exhaustive search, which examines all of the possible pixel block-pairs existing in two images. This exhaustive search is the most precise way to seek the most alike block-pair. Notwithstanding, it is too computationally demanding to be used very extensively. Hierarchical motion estimation and Fourier-Based Alignment are two examples of optimal approaches used to find the block-twin.

Hierarchical Motion Estimation In order to reduce the computational overhead, hierarchical motion estimation has been developed [4]. Each of the images in the collection is downsized into different resolutions using a downsampling factor, and an image pyramid with low-to-high resolution is established. Similarity measurement is first performed at the lowest-resolution level. All eligible blocks are compared and matched in order to obtain the commonalities. Thereafter, the motion displacement is estimated using the different locations of the same commonalities in the two images. Some prominent information may be blurred when images are downsampled. The loss of prominent information increases the risk of misalignment. Therefore, an exhaustive search is usually applied when registering images at the lowest-resolution level so as to achieve the correct alignment. When the commonalities of two overlapping images have been retrieved, and a proper motion displacement d_l has been estimated at the lowest-resolution level l, the displacement d_l is used with the downsampling factor f in order to calculate a probable displacement $\hat{d}_{l-1}$ for the next higher-resolution level *l-1*.

$$\hat{d}_{l-1} \leftarrow f d_l \tag{3.6}$$

With this predict displacement $\hat{d}_{l-1}$, the search for the commonalities is accomplished with a small area at level *l-1*. As a result, the exact displacement d_{l-1} is properly estimated. At the same time, a predict displacement $\hat{d}_{l-2}$ is computed for the high resolution level *l-2*. These steps are repeatedly executed at each level from the lowest to the highest resolution in the image pyramid. Once the exact motion displacement is obtained at the highest resolution level, the motion for the two overlapping images is precisely estimated. Therefore, the two images can be properly registered.

Since image registration is performed at a low-resolution level, the number of pixels involved in the search range is remarkably reduced. This in turn decreases the computational load. Hierarchical motion estimation does not guarantee to provide as accurate a result as in the exhaustive search, but this method is a faster way to generate approximately the same outcome. Hierarchical motion estimation has become the most popular

approach for the block matching. Due to its robustness and efficiency, the rationale of hierarchical motion estimation is feasible for image registration in order to enhance the efficiency of a visual surveillance system over a wireless image sensor network.

Fourier-Based Alignment The concept of Fourier-based alignment is based on the modulation and convolution properties of discrete Fourier transform [59]. From a signal processing point of view, discrete Fourier transform is used for the analysis of frequencies in order to resolve partial differential equations. The modulation property allows a shifted signal to transform its representation from the time or spatial domain to the frequency domain in its Fourier transform, as equation 3.7 shows.

$$F\{I(p+d)\} = F\{I(p)\} e^{-2\pi idf} \tag{3.7}$$

where p is the position of the pixel block, d is the shifted displacement, and f is the frequency of the Fourier transform.

In addition, the convolution property allows the Fourier transform to convert the convolution in the spatial domain into multiplication. This conversion significantly diminishes the amount of computation required, so that the alignment is accelerated. The Fourier-based alignment is frequently used with the function of CC or SSD to obtain the greatest reduction of computational overhead.

Many improvements and approaches have been designed for incremental refinement of either similarity metrics or alignment algorithms. In some image-stitching applications, the high accuracy of registration is needed to get an adequate output. For instance, the application may engage the neighbourhood evaluation to better the sub-pixel estimation [62], and gradient descent is applied to SSD function for computational load reduction [39]. Any combination of improved similarity metrics and advanced alignment methods can also be tailored to satisfy the specific requirements of an application, which, in turn, allows it to obtain high precision or low complexity.

3.3.2 Feature-based Registration

Unlike patch-based methods, which seek common segments by comparing the intensity deviation of pixels on block basis, feature-based algorithms explore the prominent features of each image in the collection, create a relationship based on detected common features in each image-pair, and assess the geometric transformation between the two images. Therefore, precise feature detection and adequately corresponding feature matching are vital to feature-based alignment algorithms.

3.3.2.1 Feature Detection

Feature-based algorithms are based on the extraction of distinctive common features from two images. Features are defined as identical contents in each image. They are distinct, easily extractable, and unaffected by the camera's perspective, as well as the time the image was captured. Structures such as corners, rivers or intersections are always considered to be features.

Keypoint detectors are first employed to seek "corner-like" points among images by using a Hessian matrix with translational and affine-based patch alignment [42, 55]. In order to improve the accuracy, a Gaussian weighting function is applied to the existing method to distinguish each pixel's contribution to keypoint detection [20, 26]. During the past decade, feature detectors have tended to operate independently on the resolution and the orientation of images. This has enabled the accurate feature detection among images with different resolutions or transformations. Scale-invariant Features can be obtained by searching the scale-space maxima of Difference of Gaussian (DoG) [38, 36], or by calculating Harris corner detectors [41] across a sub-octave pyramid [63]. The invariance to transformation and resolution of feature detectors provides an effective way to match images under arbitrary kinds of transformation.

Without question, keypoints are not the only type of features used for image registration. Line segments [3, 67] and distinguished areas [32, 14] are also good choices. Research on feature detectors and descriptors is evolving rapidly.

3.3.2.2 Feature Matching

Feature matching is performed after feature detection in order to figure out which feature is included in each pair of images. The function of SSD or NCC can be applied for matching features within their neighbourhood in cases where the motion around each feature is primarily translational [37, 54]. A *detect then track* approach with an affine motion model is good at tracing features in a large number of images. This approach allows for the comparison of the feature's appearance in predicted locations [55]. In situations where motion is various, or the geometric relationship among images is not clear, it is preferable that features are detected completely before being matched. This is called a *detect then match* approach [53, 7]. Mikolajczyk et al. [41] points out that Scale Invariant Feature Transform (SIFT) proposed by Lowe [38] has the best performance in general. Moreover, robust matching methods employ a variety of indexing schemes to look for the common keypoints in each image-pair [43, 44, 53].

After completely evaluating all feature correspondences, a subset of these needs to be extracted for highly precise image registration. It is profitable to obtain a suitable set of inlier correspondences in advance. RANdom SAmple Consensus (RANSAC) [18] and Least Median of Squares (LMS) [50] are two popular means with which to solve this problem. PROgressive SAmple Consensus (PROSAC), an improved version of RANSAC, is able to look for the inlier set faster by picking random samples from the most "confident" matches [12]. Experiments performed by Stewart indicate that RANSAC is highly practical, since it finds the inlier set using a minimal number of sample points [57]. In addition, a motion parameter is required for the best geometric alignment to be determined. In many cases, the SSD function is widely used to estimate the motion parameter, and its optimal versions have been devised to combat various problems [59]. For instance, the robust regression can be applied to the SSD function in order to filter outliers in the set of feature correspondences. To determine which feature-matching approach to use, we rely heavily on the feature descriptor and local motions of images. Studies on feature matching largely concentrate on the improvement of efficiency.

3.3.3 Comparison between Patch-based and Feature-based Algorithms

For the purpose of correctly registering images with a low computational overhead, we must clarify the specific requirements of a visual surveillance application. Since a surveillance application is real-time, the step of registration must be accomplished quickly, which refers a low complexity. In addition, accurate image registration is essential for effective surveillance. Furthermore, efficient observation requires not only a enlarged FOV, but rich details of a monitored region. Thus, several images are required for each mosaicking process, the number of images needed is not great. In order to figure out which type of registration algorithms is more suitable, comparisons of patch-based and feature-based algorithms, taking the image characteristics and their similarity measurement into account, are conducted.

Just as its name implies, patch-based algorithms look for the common segment of two images on a patch basis, which is usually a rectangular block. The patch pair with the minimal intensity difference is also the most similar block-twin of the two adjacent images. In general, one pair of similar blocks is sufficient for motion estimation. Feature-based algorithms measure the feature similarity using the information carried by feature points and their surrounding neighbour points. More than one common feature is required for feature matching and motion estimation.

Since human eyes are luminance-sensitive, differences between pixel intensities become the essential indicator for dissimilarity. Patch-based algorithms determine the block-twin by comparing the degree of the intensity difference between each possible block-pair within the overlapping area. Therefore, images with prominent information carried by grayscale and colours are the ideal medias for patch-based registration. In contrast, the feature description used by feature-based algorithms is any combination of the feature's intensity, curvature, orientation, shape and so forth. This more precise information is included to achieve more accurate feature matching. This is the reason

that images with distinguished details defined by contents or textures are preferred by feature-based registration.

As the aforementioned discussion, patch-based algorithms seek a pair of blocks which have a minimal intensity difference. On the other hand, feature-based algorithms extract more than one feature by matching information where intensity is not the only factor. Clearly, the complexity of a patch-based algorithm is explicitly lower than that of a feature-based approach.

Because the patch-based algorithm uses block matching to compare similarities, and only considers the pixel's intensity as the similarity criterion, patch-based algorithms are desirable when the motions of source images are translation or rotation. Source images with multiple resolutions or other complex motions registered by patch-based algorithms can introduce unimaginably huge computational overhead. On the contrary, features are defined by many other criteria besides intensity, and are invariant to scaling or transformation. Thus, feature-based algorithms are preferable to register source images that have multi-resolutions and complex transformations. Table 3.1 summarizes the above discussion.

Criteria	Patch-based algorithms	Feature-based algorithms
Compared basis	Block	Feature points and their neighbours
Compared criteria	Intensity	Orientation, texture, curvature, ...
Complexity	Low	High
Transformations of source images	Translation, rotation	Arbitrary transformation
Salient information contained in	Grayscale or colours	Textures or contents

Table 3.1: Comparison between patch-based and feature-based algorithms

In summary, patch-based algorithms are appropriate for source images that have shift and rotation motions, as well as the distinctive details carried by grayscale or colours. In this situation, low complexity can be achieved. Patch-based algorithms are suitable to obtain an enlarged FOV for real-time applications. As a result, patch-based

algorithms are beneficial for image registration in a visual surveillance application. On the other hand, feature-based algorithms are used for input images that have various transformations or scaling, as well as more salient information on textures and contents. In this situation, more accurate registration is observed. Feature-based algorithms are preferable to produce a large panoramic view of a scene such as a satellite map.

3.4 Image Composition

Once all of the images have been aligned and registered, image stitching is executed to generate the final synthesized picture. In this step, the selection of a rendering surface, the determination of the pixel contribution to the final output, and the optimal methods to diminish noticeable borders, blur and ghosting are core concerns for the production of an ideally stitched image.

A rendering surface is the presentation platform for the final output. A perspective projection onto a plane is a primitive method and always used in cases where the volume of images is not large, while cylindrical [10] or spherical [58] projections are typical selections when dealing with a large number of images. Alternatively, cube mapping is also a good choice to depict the entire globe using all the square faces [24]. Once the viewing surface is selected, a reference image should be determined as the centre of the stitched image. A geometrically central image is the usual option for the composition on a flat surface. Either the middle or the first of the input sequence of images can be the central part of the produced view if the input collection contains many images [58].

For the sake of producing a unblemished wide-angle image, visible seams, blurring or ghosting should be eliminated by evaluating the contribution of each pixel to the output view. Taking the average of each pixel in the overlapping area is a simple way, but it does not remove any of the imperfections. A median filter is employed to eliminate fast moving objects [31]. Featured average [9, 64] and Vornoi [15] algorithms correct the blurring caused by exposure differences using featured averaging with an Euclidean

deviation table. Weighted ROD vertex cover with featuring algorithm [64] removes ghosts by identifying and erasing the region of differences (RODs) in the overlapping area. Graph cut algorithms render seams invisible by cutting the image and placing a seam at the location of minimal seam penalty [16, 13]. In addition, exposure differences and misregistrations can also be balanced using either Laplacian pyramid blending [8] or Gradient domain blending [66, 48, 35].

The process of image composition not only synthesizes all source images to enlarge FOV, but also diminishes visible seams, blurring and ghosts to generate an attractive final output. The choice of image compositing approach is application-dependent, and improvements can be made according to the designated requirements of an application. However, moving objects in images, misregistration and exposure differences are always cause for concern.

3.5 Summary

A profound survey about the process of image mosaicking, including motion modeling, image registration and image composition, was discussed in this chapter. The theories and methodologies of each component of the processing were presented along with some corresponding state-of-the-art improvements. In particular, techniques of image registration were divided into two types and studied in depth. Comparisons of these two types of registration techniques were carried out. The comparison results point out that the rationale of patch-based approaches is suited for effective image registration in a visual surveillance application. In relation to the proposed algorithm discussed in the following chapters, it is necessary to point out that all developed algorithms require all images to be received and decoded completely. In conclusion, due to its widely applicable environment and low complexity, patch-based image registration tends to be a better choice for image mosaicking of a remote electronic surveillance system over a WISN.

Chapter 4

Progressive Image Mosaicking Algorithm (PIMA)

The intention of the proposed algorithm, Progressive Image Mosaicking Algorithm (PIMA), is to shorten the delay before an image's initial display caused by slow wireless image transmission. PIMA also seeks to correctly register images based on incomplete data. Furthermore, it provides a novel display of enhanced quality for users. As a result, PIMA significantly improve the efficiency of a surveillance application using a WISN and reduce the user's waiting time. Rather than apply the image mosaicking to complete images, the image mosaicking is performed during the reception of the image. This is the most distinctive characteristic of PIMA as compared to traditional algorithms. In this Chapter, PIMA's design and framework will be discussed in detail, coupled with an example of promoted applicable environments. Moreover, PIMA's four features, by which PIMA is distinguished from existing approaches, will be stressed.

4.1 Design of PIMA

As stated in section 1.1, due to the unsteady wireless connection, the slow speed of image transmission in WISNs, and the great computational load generated by the pro-

cess of image mosaicking, blending images collected from WISNs introduces a lengthy delay before the fully synthesized image is finally displayed. Moreover, the traditional top-to-bottom display of an image does not allow users to understand its contents until the whole image is presented. Improvements in image mosaicking over WISNs have focused on optimizing wireless protocols to speed up the image transmission and advancing mosaicking algorithms to lessen the overall computational cost. Progressive JPEG is an image format supported by several state-of-the-art transport protocols to accelerate image transmission. However, the overall delay is not significantly decreased because existing mosaicking algorithms use Baseline JPEG as the default image format, and do not start mosaicking until complete image data is received. Thus, the top-to-bottom display mode inherited from Baseline JPEG remains unchanged. For the purpose of shortening the overall delay when mosaicking images over WISNs, PIMA tends to display a broad Field-Of-View (FOV) image with increasing quality by using images in Progressive JPEG format. The design of PIMA is based on conclusions drawn in the survey of image mosaicking techniques stated in Chapter 3.

For the purpose of inheriting the coarse-to-fine display mode of Progressive JPEG (P-JPEG) and shortening the user's waiting time before the first display for a visual surveillance application, PIMA is designed to support the multiple scans of P-JPEG. P-JPEG's multi-scan attribute allows images to be transmitted scan by scan, resulting in an image whose quality of display is gradually improved from coarse to fine. Thus, a preview can be generated by PIMA using information contained in the early scans. Because the size of one scan is much smaller than the size of an entire image, the transmission duration of one scan is fairly shorter than that of an whole image file, so that the preview is displayed quite earlier than the complete mosaicked view. A user of a surveillance application, therefore, can obtain basic knowledge of a monitored area using the early preview in a short time. As a result, the efficiency of such an application is improved. In order to better adapt the algorithm of image mosaicking to WISNs and enhance performance, PIMA applies image mosaicking during the reception of image data. This

stands in contrast to prevailing algorithms, which mosaic images after receiving their complete data. PIMA uses three scans of P-JPEG image files to first display a merged image with an enlarged FOV as coarse and gradually becomes fine. In other words, it generate an approximate view of the wide-angle image in a short time, and continuously refining the image quality afterward. Furthermore, PIMA also offers the users the option to either wait for the refined views of the same image or to switch to the rough view of the next image based on their knowledge of the current view.

Since PIMA applies mosaicking to certain scans when receiving other image data, the risk of misregistration is increased due to incomplete information. To remedy this deficiency, PIMA invokes the concept of hierarchical motion estimation for progressive image registration. The process of PIMA is tailored to accommodate the multi-scan feature of P-JPEG files. A hierarchy of image quality level is built once PIMA decodes the three most ideal scans of the image. Latter scans bring more information to refine the display quality. PIMA executes registration first at the coarse level, and refines the quality of registration at subsequent levels as more information becomes available to correct the latent misregistration at the more coarse levels. This hierarchical refinement corrects the unexpected errors when images are registered based on partial data.

In order to adapt PIMA to surveillance applications without environmental restriction, and to maintain a low computational load, PIMA uses the rationale of patch-based algorithms for image registration, as the conclusion stated in subsection 3.3.3. It compares the intensity difference of pixel blocks between two images in accordance with a suitable similarity criterion. To improve the accuracy of the completely automatic patch-based image registration, which is based on incomplete information, PIMA registers two overlapping images by seeking a pair of blocks from both images that are not only the most similar, but also contain as much distinguishing intensity information as possible. This enhancement effectively protects the images from misregistration at the coarse level, which means that a small range search for registration refinement at the later levels can achieve a noticeable reduction in the computational overload. Figure 4.1 shows the en-

tire process of image mosaicking in a WISN using PIMA. State 1 in Figure 4.1 shows the images captured by camera sensors are encoded into six scans of Progressive JPEG versions with progressive quality. In State 2, the first scan of each image captured from different sensors are transmitted to the workstation in the control base. In State 3, PIMA generates and displays the first wide-angle view at the coarse quality once it receives the first scan for all of images from different source nodes. In the same time, the later scans of these images are being transmitted. Similarly, PIMA produces the second display at the middle quality in State 4 while other scans are being transmitted. Eventually, when the last scan of all images are received, PIMA offer the final display at the best quality level, as shown in State 5.

In summary, PIMA makes use of three separate scans of P-JPEG images so as to produce a rough image preview within a minimum time, and it hierarchically registers images at three different levels of quality in order to fix the likely misalignment. Furthermore, PIMA registers images using the principle of patch-based approaches so that a low complexity is preserved. A new criterion for similarity measurement is developed for PIMA, which successfully reduces image misregistration at the initial preview of a synthesized image. As a result, PIMA achieves an coarse-to-fine image display with correct image registration and a shorter delay. This makes a surveillance system using PIMA widely applicable to various environments.

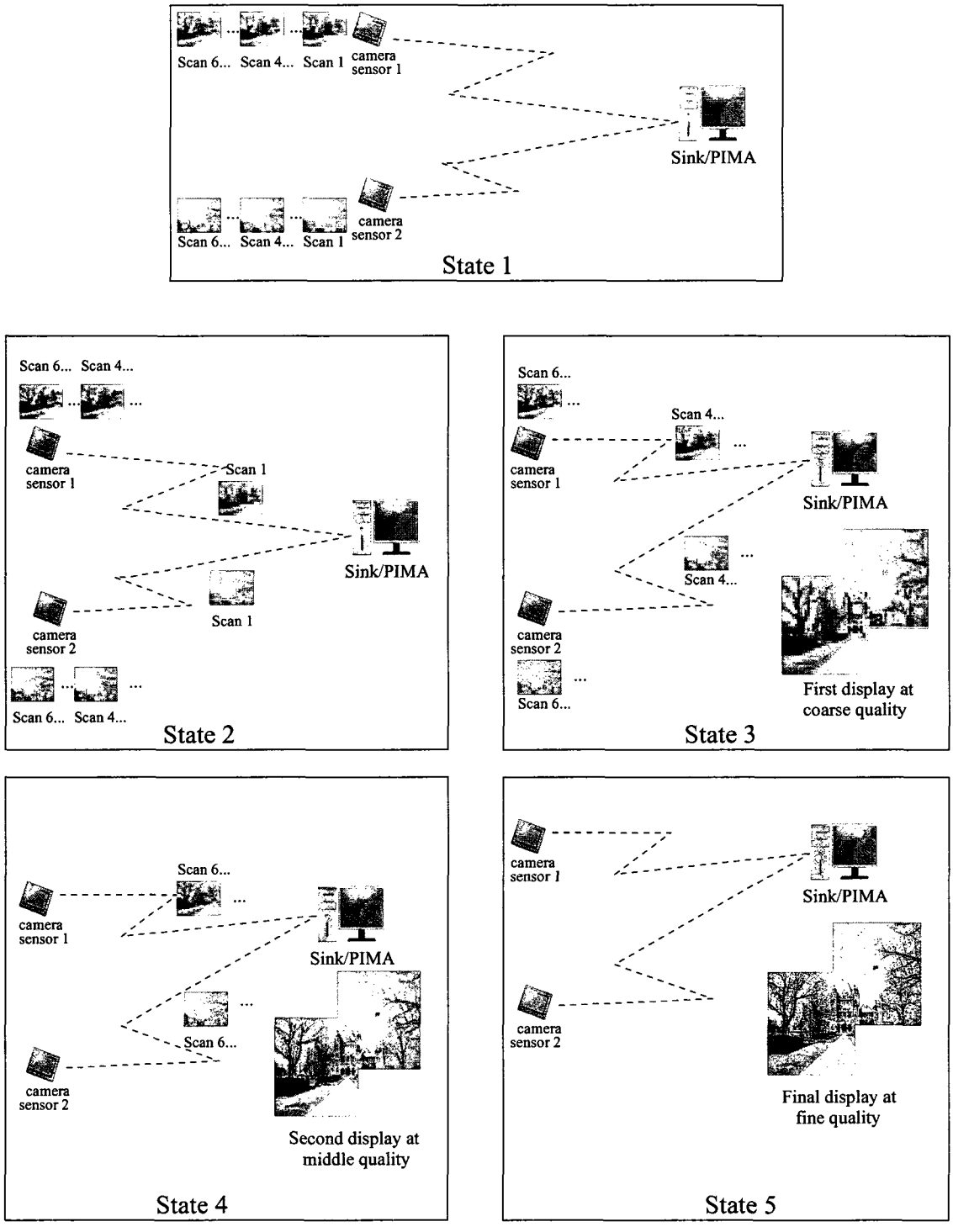


Figure 4.1: An entire process of image mosaicking in a WISN using PIMA

4.2 Applicable Environments

Wireless local area networks (WLANs) have been extensively integrated into many business practices due to their high flexibility and mobility. The information communicated over WLANs is no longer restricted to text. Audio, video, and images now can also be transmitted for multimedia applications. A real-time visual surveillance system is such an application that mosaics images collected from WLANs in order to obtain a broad FOV of a monitoring area. However, a frustrating delay is caused by transmitting large image files over unreliable wireless connections using insufficient bandwidth. This delay prevents surveillance applications from developing further.

PIMA is designed for image mosaicking using a group of images encoded in the form of P-JPEG, and transmitted from different camera sensors to applications over WLANs. Since PIMA is able to progressively improve the quality of display with very short delay, all of the aforementioned disadvantages become insignificant, and the number of relevant environments increases. PIMA is of great benefit to budget-constrained applications, or environments in which using a wireless network is the only practical choice. It also shortens the delay before the approximate image preview is available, and more camera sensors can be distributed at multiple locations to broaden the monitored area. More images can be taken and sent to PIMA in order to render a larger virtual environment. Therefore, wireless remote surveillance can be popularized and expanded in all possible civilian applications. Figure 4.2 illustrates an instance of a remote electronic surveillance system over a WISN at the site of a fire.

4.3 Framework of PIMA

PIMA is designed to mosaic images using P-JPEG's scans and to eventually shorten the delay before the first display of a merged image. Since each scan of a P-JPEG image only contains segmental information, accuracy of image registration suffers from these incomplete image data. Therefore, PIMA concentrates on improving the accuracy of pro-

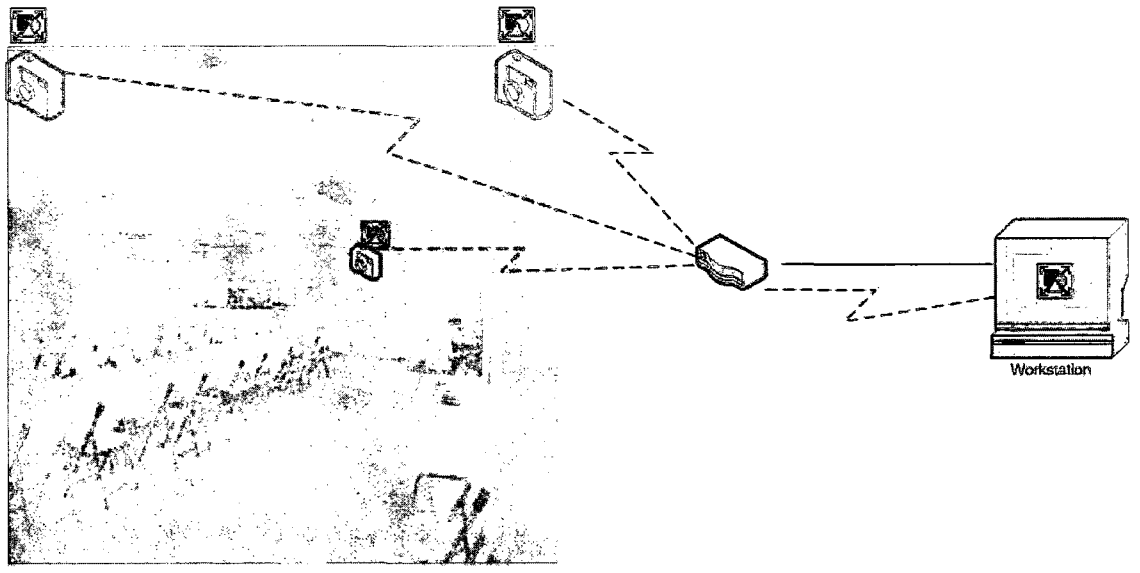


Figure 4.2: Remote surveillance using PIMA

gressive registration. The concept of PIMA is based on the combination of the multi-scan attribute of P-JPEG and the idea of hierarchical motion estimation, because P-JPEG provides certain scans for progressive mosaicking and hierarchical motion estimation has proven to be highly efficient for image registration.

Instead of applying image mosaicking after receiving the complete images, PIMA chooses three scans of P-JPEG images according to Scans Preference while receiving the scan stream of these images. The selected scans are then decoded and categorized into three levels of quality, *Coarse Level, Middle Level,* and *Fine Level.* Image mosaicking is first executed at the coarse level, which contains fractional information of each image. While registering each pair of images at this level, a Richer Information and Likelier (RIL) block-pair is sought to enhance the quality of image registration, and an Image Blocking Search (IBS) is carried out to accelerate the RIL block-pair search. Due to its low complexity, a variation of the SAD function is developed to evaluate and obtain such a RIL block-pair. After registering the two images, the merged preview at low quality is then delivered for a user's first glance. Because the information derived from images at the coarse level is incomplete, the next step is to refine the quality of the image

registration as well as the quality of display. Based on the information carried by the RIL block-pair at this primary level, a Small Range Search (SRS) can be repeatedly used to improve the accuracy of image alignment at subsequent levels of quality. Therefore, the RIL block pairs corresponding to the middle level and fine level can then be attained. According to the location of the associated RIL block pairs, images at the middle and fine levels can be consequently stitched together to produce higher quality displays with a magnified FOV.

To further clarify the following explanation, a few assumptions applicable for the following discussion should be declared.

- Scans of different images are synchronized before mosaicking begins.
- The step of motion modeling is completed, whereby geometrical errors and exposure differences have been corrected.
- A pair of overlapping images k and $k+1$ is waiting to be registered.

4.4 Features of PIMA

In order to accomplish correct registration using incomplete image information, PIMA has some advanced features that enable it to deal with multiple scans of P-JPEG files. The essential features of PIMA, which differ from those of other existing algorithms, consist of Scans Preference, Image Blocking Search (IBS), Richer Information and Likelier (RIL) block pair, and Small Range Search (SRS). Scans Preference refers to the selection of scans used to construct a hierarchical image quality structure for progressive mosaicking. IBS reduces the number of eligible block pairs in two images so as to facilitate similarity comparison. The RIL block pair is a newly-developed modification of the SAD function, and a new similarity criterion for image registration improvement. SRS is a slight shift search to gradually enhance the quality of image registration. The following subsections describe these four features in depth.

4.4.1 Scans Preference (SP)

According to the premise of hierarchical motion estimation [4], the number of quality levels in the hierarchy is crucial to the algorithm's efficiency; specifically, the amount of computation required and the precision of image alignment. Obviously, if levels are too few, the time needed to generate the first view cannot be reduced notably, or images cannot be aligned precisely. On the other hand, too many levels not only introduce a great deal of computation, but also cause the principal information to be blurred before recognition is possible. A three-level image hierarchical structure is promoted as the optimal means of achieving the best result while also ensuring a minimal computation.

The total scan-number of a P-JPEG file depends on the default configuration of an image library or the specific requirements of applications. Colourful images have more colour components than grayscale images, so more coefficients must be grouped into more scans. The first scan can contain either the DC coefficient alone, which is the first coefficient of an 8×8 DCT block, or both the DC coefficient and the first several AC coefficients. All other AC coefficients are divided into several clusters and grouped into different scans.

Of all the scans in an image file, any three can be designated to form the image hierarchy. The problem lies in deciding which three scans are best suited. Arbitrary selection could lead to misregistration, distortion, or unacceptably harsh quality presentation. The decision depends on the total number of scans and on the content of each scan, as the ultimate goal is to generate a better display with minimal delay.

- **Coarse Level:** Since the DC coefficient implies the average value of the image data in a DCT block, it carries the most common information and represents the primary characteristics. DC coefficients are the only vital segments needed to provide a harsh but recognizable view of an image. Additionally, each DCT block has a single DC coefficient, so that little space is required to transmit it. As a result, the first rough view can be obtained in a relatively short time. Therefore,

PIMA limits the first scan to include only the DC coefficient of each DCT block, and assigns the first scan at the coarse level of the image hierarchy.

- **Middle Level:** The middle level can be any scan after the first scan and before the one allocated to the fine level, depending on the requirements of the application and the settings of the image library. The earlier scans can provide a fast second rendering, but this rendering does not significantly help to improve the quality of registration and display. The latter scans show a remarkable improvement in quality, but at the cost of a long transmission delay. Conventionally, the median scan seems the best choice for the middle level. However, in certain cases, the scan directly after the median one is more attractive; the more information it contains, the better the image quality and the higher the accuracy of image registration. Either of these two scans presents a valid option for the middle level of the image hierarchical structure.

- **Fine Level:** There is no doubt that all scans will be acknowledged eventually. Receiving the final scan allows the composition of the entire image using the complete information, and resulting in a fine-quality presentation at this last stage. There is no better choice than the final scan to be designated as the fine level.

Figure 4.3 demonstrates an example of a quality hierarchy decoded from three scans of an Progressive JPEG image.

4.4.2 Image Blocking Search (IBS)

To decipher the similarity-degree in each pair of images during image registration, patch-based approaches use a search algorithm to look for the likeliest block-pair, as evaluated by one similarity criterion, such as SAD. The overlapping status of these two images is then extracted based on the location of the likeliest block-pair. A great number of block pairs in both images are qualified for similarity comparison, thus, an effective search algorithm is crucial to accelerate image registration.

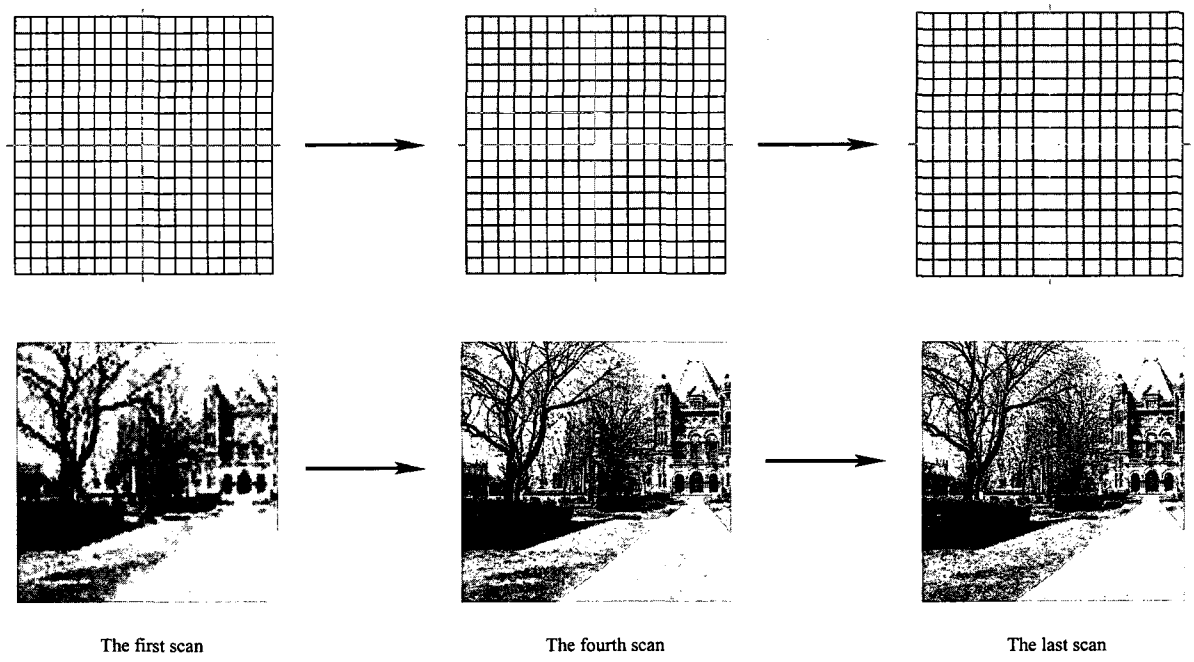


Figure 4.3: Three quality levels of an image hierarchy

For each pair of overlapping images, an exhaustive search is the most intuitive approach to find a pair of blocks with the minimal SAD value. This is defined as the likeliest block-pair. As its name implies, the exhaustive search automatically checks all the eligible block pairs using each pixel as the starting point within both images. This is the most accurate method to search for the target block-pair. However, the exhaustive search generates a large amount of computation due to the great number of block pairs involved. For example, if the resolution of the two images is $m \times n$, the number of all eligible block pairs is m^2n^2. The computational overhead is so huge that it is impractical for use with image mosaicking applications. Hierarchical motion estimation is noted for its efficient search, yet it also applies an exhaustive search at the level with the lowest resolution so as to avoid misregistration.

To accelerate the search for the likeliest block-pair, PIMA employs an Image Blocking Search (IBS) in order to reduce the number of the eligible block pairs of two images. IBS splits one of the two images, i.e. image k, into a series of pixel blocks of an appropriate

size. The size of the blocks is flexibly predefined with careful consideration. Blocks of too small a size will not offer sufficiently prominent information. Intensity differences among these blocks are inconspicuous. Contrastingly, blocks of too large a size contain too much information, some of which is useless. The salient information in large blocks can be blurred when averaging the pixel intensity if too much irrelevant information is present. More fake likeliest block pairs are detected in either case, and the probability of misregistration becomes greater. Experiments have been conducted to determine a proper size for the blocks using in IBS. It has been observed from experimental results discussed in Subsection 5.2.2 that the size of pixel blocks is associated with the image resolution. In most cases, a block in approximately one percent of the image size is the most suitable because it contains exactly enough distinct information for precise similarity measurement.

After blocking image k, the number of eligible blocks is reduced to about 100, and the number of possible block pairs for the SAD calculation is exponentially decreased accordingly. As a result, the search for likeliest block-pair is accelerated, and the computational overhead is reduced.

4.4.3 Richer Information and Likelier (RIL) Block Pair

To precisely align all the images at the coarse level, a suitable similarity metric is required to measure the intensity difference of each eligible block-pair. SSD, SAD, NCC, and their enhancements, as stated in Subsubsection 3.3.1.1, can accurately evaluate luminance deviations for all eligible block pairs. Unfortunately, these approaches have the prerequisite that the image data be complete in order for a satisfactory performance to be obtained. PIMA performs mosaicking based on the partial information carried by designated scans. In this way, PIMA offers a contrast to prior methods. With this knowledge in mind, a new similarity evaluation method for PIMA using this segmental data is outlined as follows.

The only factor that existing methods take into account for similarity measurement between a block-pair is the intensity difference representing the degree of similarity. The pair of blocks which has the smallest value of SAD is treated as the most similar pair, even though the blocks in this pair are not necessarily the same. These blocks are fake likeliest block-pairs. This illusion becomes more frequent when the SAD function is applied directly to PIMA. PIMA registers images using their first scans, which only contain DC coefficients that represent the average of DCT blocks. Therefore, the intensity differences between blocks at the coarse level are not explicit enough. The risk of obtaining fake likeliest block-pairs increases. Other information should be taken into account for similarity assessment.

It is observed that if the likeliest block-pair is selected manually by a person, so as to increase the precision of image registration, this block-pair tends to contain the most striking features and the greatest contrast to its neighbours. That is, information carried by this block-pair is abundant and various. For example, blocks in this pair could contain the vertex of an object, or the interface of two objects. The intensity of pixels within this block-pair could vary greatly. In fact, the sum of intensity distance between each pixel and the average of the block, which is the self-intensity deviation, is great. This is not the case with the block-pair, in which the likeness is the only concern. The manual method indicates that the amount of information contained by the pixel block should also be a factor in block matching.

PIMA imitates the manual method by targeting a pair of pixel blocks in which the blocks are not only similar to each other, but also hold as much distinctive information as possible, and thus improve image registration. In other words, PIMA aims to find a pair of blocks that have not only the smallest SAD between them, but also the greatest SAD within themselves. This sort of block-pair is identified as a Richer Information and Likelier (RIL) block-pair. There are not many RIL block pairs existing in two overlapping images. Some blocks have feature-like information, but they are not in an overlapping area, which means they cannot match any block of another image. Some

blocks have similar blocks in another image, but they are not truly matched. These block-pairs are fake likeliest block-pairs and are probably different parts of repeated patterns. Therefore, if a rich-information block in image k has its most similar block in image $k+1$, this RIL block-pair is truly matched and reliable for image registration. Taking into account the image's distinct information for similarity estimation successfully separates actual likeliest block-pairs from feigned ones, so that the risk of misregistration caused by partial image information is significantly diminished at the coarse quality level.

Prior to further examination, some denotations must be addressed.

- The size of the pixel block is set to be $m \times n$.
- The size of the images is unique, whereby H denotes the height and W denotes the width.
- $I(i,j)$ denotes the intensity value of a pixel at the position (i,j) in the current block.
- (p,q) is the position of the current block in image k, and (x,y) is the position of the current block in image $k+1$.

The following steps explain the process of obtaining the RIL block-pair in PIMA at the coarse level.

1. Split image k into a number of pixel blocks with a predefined size $m \times n$ using Image Blocking Search.
2. For each pixel block of image k, calculate the SAD of the block (Self_SAD) by summarizing the intensity differences between each pixel and the average intensity level of the block. The block with the greatest Self_SAD is the one containing the most prominent information about image k.

$$Self\_SAD_k(p,q) = \sum_{i=1}^{m}\sum_{j=1}^{n} |(I_k(i,j) - \frac{1}{mn}\sum_{i=1}^{m}\sum_{j=1}^{n}(I_k(i,j)))| \qquad (4.1)$$

3. In order to find its best match in image *k+1* for each pixel block of image *k*, compare this block to all eligible same-size block of image *k+1*. The intensity difference between this block and each of the qualified blocks of image *k+1*, which is Inter-block SAD (ITB_SAD), is calculated. The block of image *k+1* with the smallest ITB_SAD (MIN_ITB_SAD) is the best match for the block of image *k*.

$$MIN\_ITB\_SAD_k(p,q) = \min_{1\le x\le H, 1\le y\le W}\left(\sum_{i=1}^{m}\sum_{j=1}^{n}|(I_k(i,j) - I_{k+1}(i,j))|\right) \quad (4.2)$$

4. For each pixel block of image *k*, take its Self_SAD divided by its MIN_ITB_SAD. The block with the maximal rate (RIL_Rate) of Self_SAD over MIN_ITB_SAD is the one that has greater Self_SAD and less MIN_ITB_SAD. To put it another way, this block of image *k*, with its likeliest match in image *k+1*, offers as much information as possible. This block-twin is the expected RIL block-pair.

$$RIL\_Rate_k(p,q) = \max_{1\le p\le \frac{H}{10}, 1\le q\le \frac{W}{10}}\left(\frac{Self\_SAD_k(p,q)}{MIN\_ITB\_SAD_k(p,q)}\right) \quad (4.3)$$

In PIMA, the similarity criterion is no longer simply the intensity's SAD of two blocks, but rather, the enhanced version of it. The Self_SAD, representing the amount of salient information contained in the block, is used with MIN_ITB_SAD to obtain the RIL block-pair. The combination of great salient information and small intensity difference filters out the fake block pairs that do not actually match despite their small SAD values. The RIL block-pair successfully amends the misregistration due to incomplete image information.

4.4.4 Small Range Search (SRS)

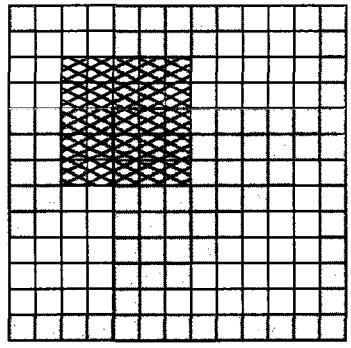
Even though image registration has made progress by seeking the RIL block-pair, it may not be precise enough. Due to the fact that only partial information is available for registration, minor errors probably exist at the coarse level. These errors may not be

noticeable in the initial preview of a merged image, however, they affect the result of seamless integration at other two levels with better quality. After image mosaicking has ceased and a blurry panoramic image has been exposed, refinement is necessary at the middle and the fine levels for display and registration.

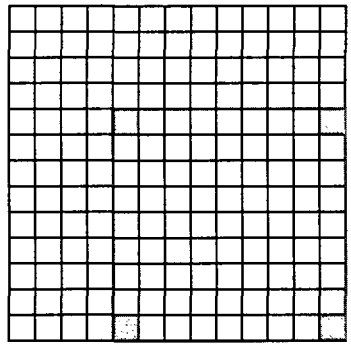
The coarse-level images are decoded from the first image scan that has only DC coefficients. These DC coefficients contain the most significant information of DCT blocks, so that the risk of misregistration is not great. Furthermore, the use of the RIL block-pair also significantly reduces the chance of misrecognition at the coarse level. The refinement search can therefore be constrained to a small scope around the original position, thus avoiding high computational overhead.

Small Range Search (SRS) is engaged for the image registration refinement according to the information presented by the RIL block-pair at the previous quality level. As image k shown in Figure 4.4 illustrates, the RIL block at the coarse level (C_RIL) obtained by performing registration is surrounded by the red lines. This C_RIL block is the temporary RIL (T_RIL) block at the middle level, and its position determines the search range of SRS at the middle level. Because resolutions of all images in the hierarchy are unique, the location of the T_RIL block at the middle level is the same as the position of the C_RIL block at the coarse level. In Figure 4.4, the position of the T_RIL block surrounded by blue lines at the middle level is the same as the position of the C_RIL block surrounded by the red lines at the coarse level. SRS shifts the T_RIL block at the middle level of image k two steps in all directions, which means that the T_RIL block traverses all 24 positions surrounding it, plus its original position.

As Figure 4.4 shows, the 25 pixels marked with cross patterns are possible starting points of the T_RIL block. These 25 pixels represent all possible T_RIL block positions for SRS at the middle level. Similarly, SRS obtains the location of T_RIL block at the middle level based on the location of C_RIL block of image $k+1$, and shifts the T_RIL block in the same way to get 25 eligible T_RIL blocks that in total consist of the C_RIL block and its two-step neighbours. In each of these positions, the T_RIL block on image



Middle Level



Coarse Level

Figure 4.4: SRS of the RIL block at middle level

k is compared to all the eligible T_RIL blocks on image *k+1* by evaluating the value of ITB_SAD. In other words, the original RIL blocks will traverse all of the 25 likely positions of their corresponding images and will be checked for their similarity. The pair of T_RIL blocks with the minimum value of ITB_SAD is updated to reflect the new RIL block-pair at the middle level (M_RIL). The block at the middle level surrounded by red lines in Figure 4.4 is the M_RIL block. If there is a slightly inaccurate misregistration at the coarse level, it will be corrected by SRS at the next level. After the M_RIL blocks corresponding to image k and *k+1* are obtained, image compositing at the middle level can then be processed according to the location of the M_RIL block-pair. A fully synthesized image of better quality is then generated for improved presentation.

Eventually, SRS is performed again at the fine level using the information contained in the M_RIL block-pair at the middle level. SRS then explores a new RIL block-pair (F_RIL) to produce the final merged image. By applying SRS repeatedly at the two finer levels, minor misalignments are fixed hierarchically. Figure 4.5 demonstrates the concept of the Small Range Search.

The RIL block pairs at all three quality levels tend to be the same in most cases, and so, it seems redundant to use SRS for registration amendment. Nevertheless, SRS is a fundamental step because the negligible flaws at the coarse level may become apparent at the refined levels. The final wide-angle image can show evidence of seams or offsetting if SRS is not used. Therefore, the use of SRS ensures the precise image registration at the final display, and no slight offset remains.

4.5 Summary

In order to improve the efficiency of a visual surveillance application, release the environmental constraint, and shorten the delay due to slow image transmission over a WISN, PIMA adapts the traditional process of image mosaicking for Progressive JPEG. This allows PIMA to generate multiple versions of an image for display. A rough but recognizable preview is generated first, and the display is gradually refined later. This quality-progressive display and the early preview not only shorten a user's waiting time, but also allow the user to obtain a basic comprehension of a monitored region. To sum up, the design and the framework of PIMA have been addressed in this chapter, and improvements have been devised to diminish the number of the opportunities of misregistration. Scans Preference can be applied to hierarchically partial image decoding and is the foundation of correct image alignment and progressive quality. Image Blocking Search reduces the number of eligible block pairs to accelerate the search for the likeliest one. Richer Information and Likelier block-pair, which is an evolved version of the Sum of Absolute Difference, exceedingly enhances the final result when registering images.

Small Range Search betters the image alignment at the finer levels to avoid potential distortions created by a slight shift at the coarse level. These four features greatly contribute to perfecting the image registration. Moreover, these features help avoid excessive computation by limiting the number of quality levels involved, the number of block pairs needed for similarity comparison, and the search area of SRS. Furthermore, they shorten the delay before the first harsh view by defining the content of the coarse level. Scans Preference, Image Blocking Search, Richer Information and Likelier Block pair and Small Range Search form the backbone of PIMA.

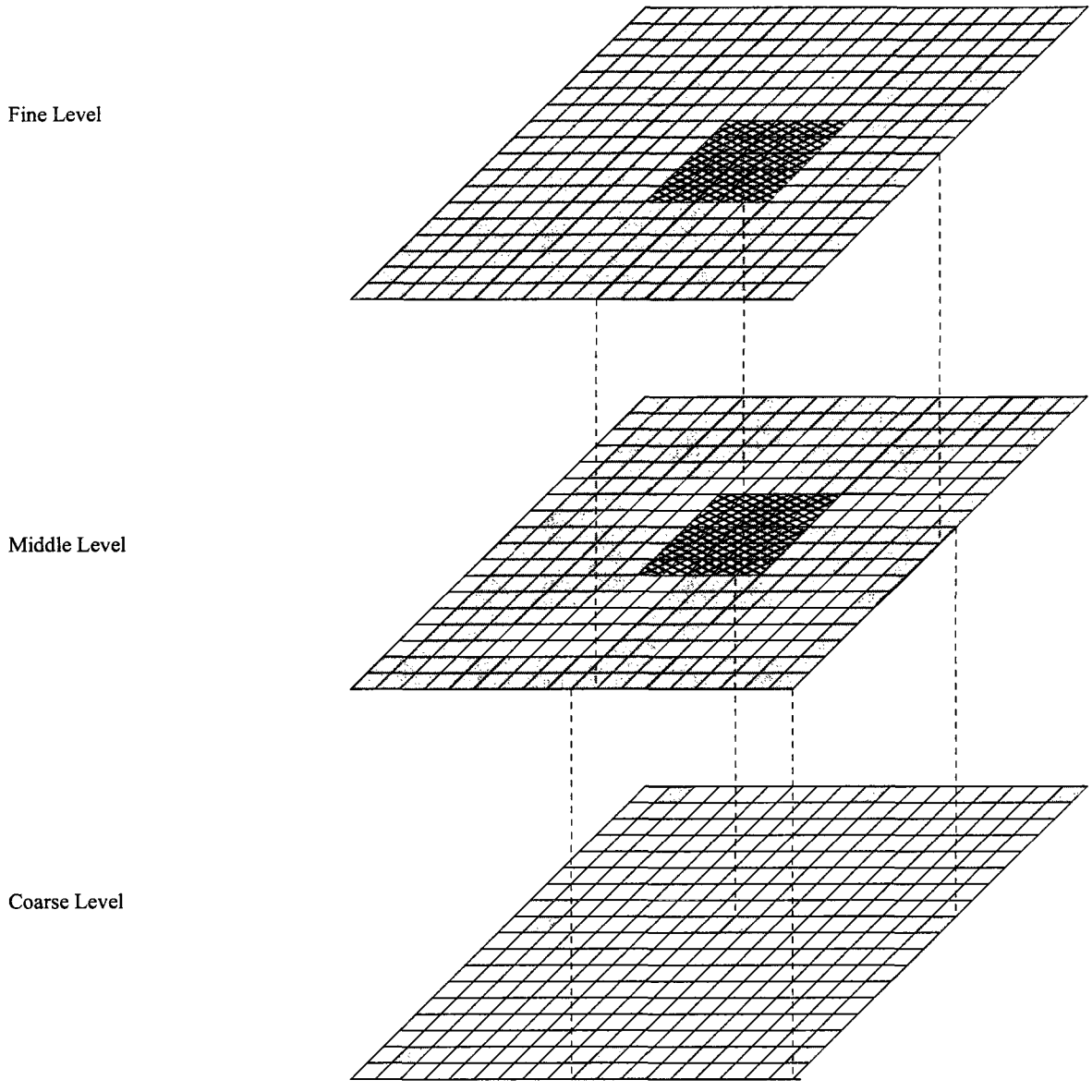


Figure 4.5: Small Range Search applied at two finer levels of quality

Chapter 5

Implementation of PIMA

In the previous chapter, PIMA's design and framework have been introduced in depth, and its four essential features have been theoretically expounded. In this chapter, the implementation of PIMA, including its workflow and its four features will be discussed in detail, as well as problems resolved during its implementation.

5.1 Design of Implementation

For the sake of implementing PIMA, the overall workflow must be designed with details in advance. Firstly, in order to offer a user a rough and early preview in a visual surveillance application, PIMA designates Progressive JPEG (P-JPEG) as a preferable image format, and inherits its feature of coarse-to-fine display. Image stitching is performed at each quality level with images decoded from certain scans of P-JPEG. Images captured by camera sensors must be encoded into P-JPEG format before they are sent. Scans of the P-JPEG image must be recognized by a wireless image transport protocol, and the anticipated scans must be retrieved and synchronized from all scans among different images before they are mosaicking. Figure 5.1 shows the flowchart of Scans Preference.

For the purpose of reducing the complexity, the process of Scans Preference must be integrated into the image decompression process at a sink node. A suitable wireless

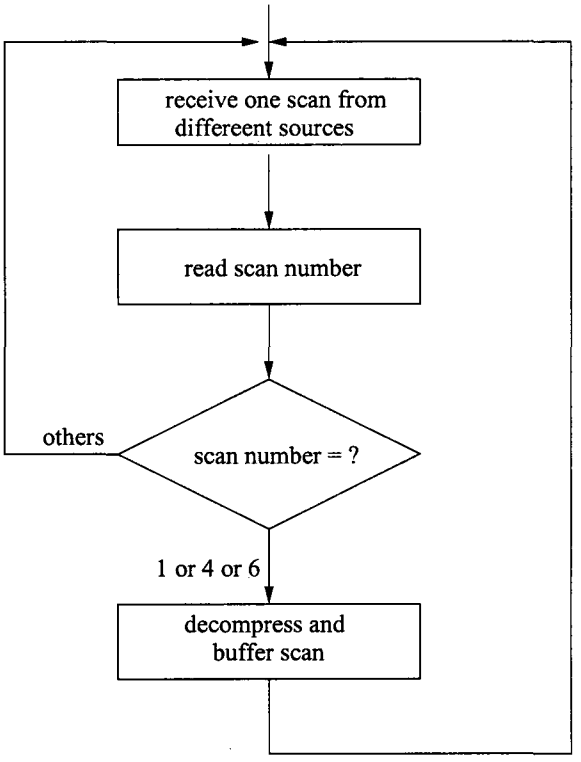


Figure 5.1: The flowchart of Scans Preference

image transport protocol is required to collaborate PIMA's implementation. Reliable Synchronous Transport Protocol (RSTP) is a newly-developed protocol that offers the implementation needed to transmit P-JPEG files across WISNs [6]. The incredible attraction of RSTP is that it provides some straightforward methods for customizing the use of scans from P-JPEG files, so that multiple scans can be identified and separated for Image-Based Rendering applications such as PIMA.

After the scans have been synchronized and decoded, PIMA is then executed as per the following description. Figure 5.2 shows the workflow of PIMA.

1. According to Scans Preference, which is discussed in Subsection 5.2.1, PIMA chooses the first, fourth and sixth scans of grayscale images to produce images at three levels of quality. When receiving scans, PIMA first checks the scan number. If the scan number is not 1, 4 or 6, PIMA skips these scans and receives the next scans.

PIMA repeats this step until one of each of the required scans has been received.

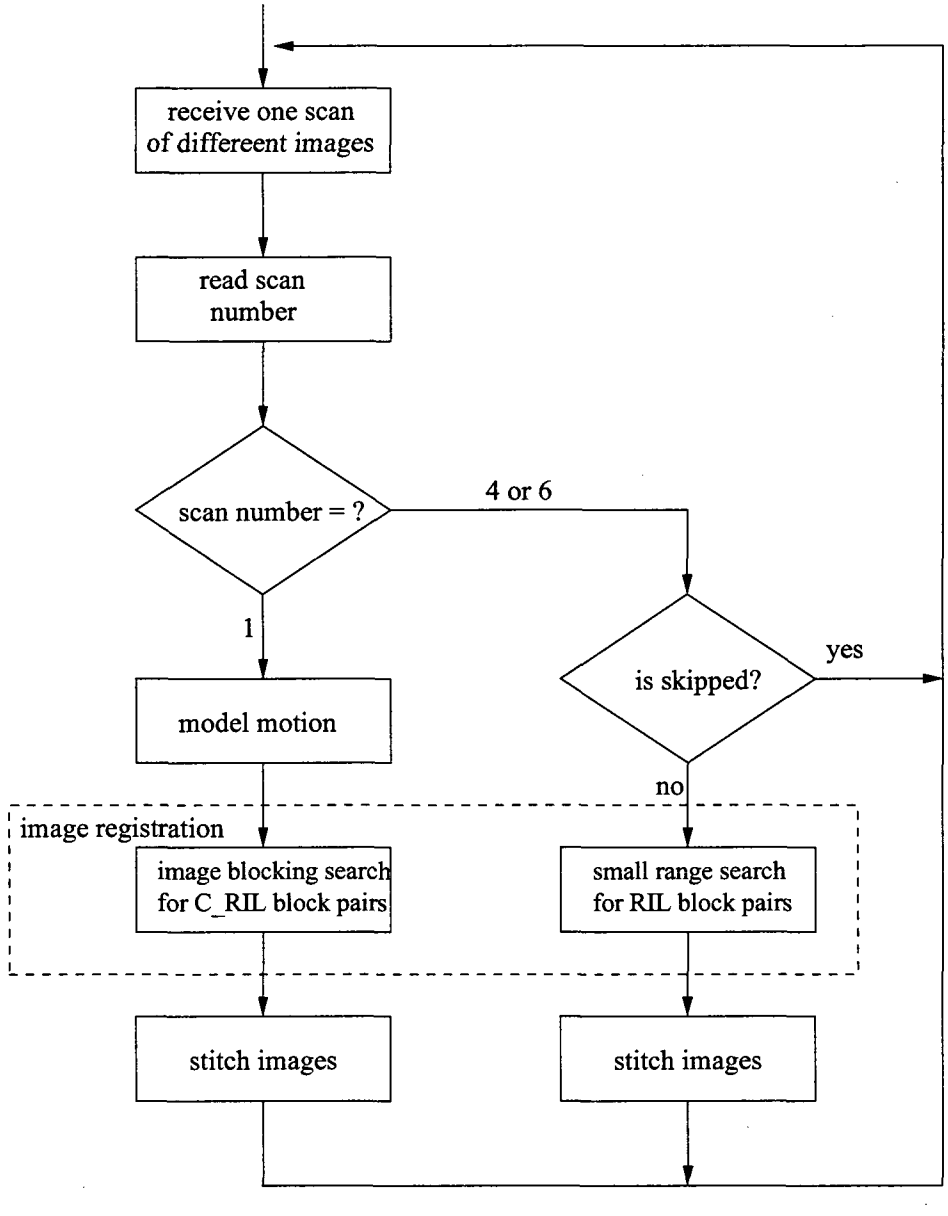


Figure 5.2: The workflow of PIMA

2. If the received scans are labeled as scan 1, PIMA decompresses them into RGB images and buffers these images. Thereafter, these RGB images are grouped as overlapping images at the coarse level of quality.

3. Motion modeling is then applied to establish mathematical relationship of pixel coordinates for each pair of images.

4. For each pair of images in the group, PIMA applies Image Blocking Search to divide one image into a number of blocks of a predefined size. PIMA then searches the RIL block-pair.

5. PIMA stitches images together based on the location of RIL block pairs, and produces an assembled image with a broad FOV. PIMA also saves the location of the RIL block pairs for refinement later on.

6. If the received scans are labeled as either 4 or 6, PIMA determines whether the user chooses to skip the refined view or not. If the skip is chosen, PIMA disregards the scans, releases the information of RIL block pairs saved beforehand, and reverts back to step 1 to receive the next scans.

7. If the user does not choose to skip the refined view, PIMA decompresses scans 4 or 6 into a group of overlapping RGB images at the middle or fine level.

8. PIMA retrieves the location of RIL block pairs from the previous level, and applies Small Range Search to get new RIL block pairs at the current level.

9. At either the middle or fine level, PIMA merges the images based on the location of the corresponding RIL block pair, and produces the wide-angle image with increasing quality. PIMA only saves the information of RIL block pairs at the middle level before returning to step 1.

The implementation of PIMA employs only grayscale images for mosaicking so far due to the fact that luminance is a significant component of image processing. In most cases, a colour image is transformed from RGB colour space into YCrCb colour space, so that the luminance component can be separated from the chroma components for subsequent processing. Nevertheless, it is easy to extend PIMA to support colour images

in a similar way to grayscale images. For each possible block-pair of two colour images, intensity difference can be calculated for each colour channel in the RGB colour space. These intensity differences across all of three colour channels can then be summarized and compared so as to obtain RIL block pairs at each quality level. The same hierarchical registration and compositing approaches can, therefore, be applied to produce the broad-angle colour views that will continually improve in quality.

5.2 Implementation of PIMA

Due to the multiple levels of quality in PIMA's display, the process of registering images offers contrast to traditional algorithms. More misregistration and computational overhead are introduced when aligning images based on incomplete information. PIMA focuses on advancing the precision of image registration using a hierarchical process. Thus, PIMA's implementation concentrates on the process of image registration.

In general, cameras in a surveillance surveillance application are mounted on wireless sensors and have overlapping FOVs with their neighbour, panning horizontally to capture images. Translation and projective are the primary 2D transformations of the collected images. Focusing on the implementation of registration, PIMA makes the assumption that translation is the only 2D transformation. Motion modeling therefore is simplified to the conversion of the pixel coordinates of two images. From a surveillance point of view, only a few images are required to create an enlarged FOV. A flat surface is adequate for image composition. PIMA chooses the first received image as the reference, and then projects other images according to the pixel coordinate system of the reference image. The flat compositing is not only straightforward, but also maintains the straightness of lines after the projection of the other images, which is always an attractive characteristic for image mosaicking.

The process of image registration in PIMA is divided into three steps, an initial alignment is performed at the coarse level, and enhancements are made at the other

two levels. The implementation of registration is discussed in the following subsections according to the four essential features.

5.2.1 Scans Preference (SP)

Since PIMA mosaics images using three scans of each P-JPEG file, these three scans should be retrieved in isolation from the whole file. To diminish the computational overhead, it is ideal that only these three scans are decoded for PIMA, because all other scans can be ignored. Thus, Modifications for Scans Preference are added to RSTP before received images are sent for decompression at the sink of a WISN.

RSTP [6] is an image transport protocol designed to suit the progressive transmission of P-JPEG image files. RSTP compresses an RGB image into a *.jpg* file with the progressive mode at sender nodes, and decompresses the *.jpg* file one scan at a time at the receiver end. Additionally, a number representing the scan sequence of a P-JPEG file is inserted into the RSTP package header. Therefore, Scans Preference of PIMA can recognize the three designated scans once the sink obtains the data packages. Scans Preference then send the scans through the process of decompression. Furthermore, RSTP uses the scan number to synchronize scans from different camera sensors. The next scans are not sent out until the current scans from all camera sensors have been acknowledged by the sink. Thus, Scans Preference of PIMA can correctly gather the scans with the same number from different images, which means the designated scans with the same quality from different images are grouped together for mosaicking afterwards. The scan number and the scan synchronization facilitate Scan Preference of PIMA to collect the same designated scan from different image files for mosaicking.

RSTP employs LIBJPEG to manage image compression and decompression using the progressive mode of the operation. LIBJPEG is an image library innovated by the Independent JPEG Group (IJG) for JPEG image compression, and it provides the encoding and decoding functions for P-JPEG. By default, LIBJPEG encodes a grayscale image into a P-JPEG file using six scans (ten scans for a colour image), in which the first

scan contains only DC coefficients. PIMA takes this default setting and identifies three scans of each image in order to build the image hierarchy. According to the discussion of Scans Preference in Subsection 4.4.1, scan 1, scan 4 and scan 6 are the three ideal scans. Therefore, as PIMA receives the data stream of a P-JPEG image, instead of decompressing the whole file, PIMA identifies the scan number and saves the preferable scans for later processing.

In order to improve the efficiency of a visual surveillance application, PIMA offers the option that allow a user to skip the following refined version(s) of an image if he satisfies the current view. Scans Preference uses a Boolean variable "skip" to indicate the user's choice. When the user chooses to skip the scans, the variable is set to "TRUE"; otherwise, it is set to "FALSE". After PIMA received scans labeled 4 or 6, Scans Preference checks the Boolean variable before sending the 4th or 6th scans for decompression. The designated scans will be discarded if "skip" is TRUE, and no decompression will be applied to these scans. The designated scans will be decompressed only when "skip" is set to FALSE, which means that the user likes to view the refined version(s). This scan-skipping option reduces the computational overhead, and accelerates the overall process of mosaicking. Thus, the efficiency of a surveillance application is improved.

When receiving the data stream, PIMA checks the scan number from the package header and distinguishes its anticipated scans. When the expected scans are received, Scans Preference of PIMA first inspects whether or not a user choose to skip these scans. If the user chooses to skip them, PIMA discards these scans, and then receives scans with the next scan number. If the user chooses to save these scans, PIMA sends them to decoding process and stores the decoded image data in separate buffers for further mosaicking. Data of other scans is discarded. PIMA uses the method `decompress_JPEG_mem()` from RSTP to decompress scans from Progressive JPEG to RGB in memory. PIMA receives and buffers scans of next quality level during mosaicking, so that the next scans will be ready when the previous compositing is complete. If the next scans are not ready when the previous composition is finished, mosaicking will not start until all expected scans

from all camera sensors have been received completely. Figure 5.2.1 is pseudo-code of Scan Preference.

```
for (each sensor){
  switch (scan_number){
    case 1:
      decompress_JPEG_mem();
      mosaic_coarse();
    break;

     case 4:
       while (!skipped) {
          decompress_JPEG_mem();
          mosaic_finer();
       }
    break;

     case 6:
       while (!skipped) {
          decompress_JPEG_mem();
          mosaic_finer();
       }
    break;
}
```

Figure 5.3: Pseudo-code of Scans Preference

5.2.2 Image Blocking Search (IBS)

The goal of Image Blocking Search is to accelerate the speed without downgrading the accuracy of image registration at the coarse level. In each pair of images, one is divided into a series of blocks of a pre-defined size so as to reduce the number of blocks available for similarity evaluation. The search for the RIL block-pair is then accelerated, while the computational load is reduced.

Image registration of PIMA is patch-based, or it evaluates the degree of similarity on a block basis. A block must be defined before image registration can be performed. An exhaustive search used in traditional mosaicking algorithms defines the size of such a block, and checks all eligible block pairs using each pixel as the starting point within both images. In contrast, IBS splits one of the two images into multiple blocks with a pre-defined size, and examines all available blocks of another images for each of these blocks. The question is how to determine the size of this block. As discussed in Subsection 4.4.2, an oversize or undersize block increases the possibility of the incorrect image registration. A block of a perfect size should have sufficient identifying information and minimal useless data. Experiments were carried out to obtain the suitable size of the block. Image registration is applied to two image groups of two different resolutions. Each group consists of fifty pairs of adjacent images. The common resolutions of images captured by wireless camera sensors are 352×288 of CIF (Common Intermediate Format) or 640×480 of VGA (Video Graphic Array) [1]. Thus, the resolution of one image group is 352×288, and the resolution of the other group is 640×480. In addition, the sizes of the pre-defined block are set at five levels. Table 5.1 and 5.2 are experimental results of misregistration rates of the two groups of images.

Experimental data show that minimal misregistration occurs when the block size is approximately one percent of the image size. The number of misregistration will increase when the block size deviates from this value. As a result, the most desirable block size is set to one percent of the image.

Once the block size is determined, IBS splits one of the overlapping images into

Block Size	Number of Registration	Number of Misregistration	Misregistration Rate
10×10	50	3	6%
20×20	50	2	4%
30×30	50	1	2%
40×40	50	2	4%
50×50	50	4	8%

Table 5.1: The misregistration rate comparison of images with CIF resolution

Block Size	Number of Registration	Number of Misregistration	Misregistration Rate
40×30	50	3	6%
50×40	50	1	2%
60×50	50	1	2%
70×60	50	3	6%
80×70	50	5	10%

Table 5.2: The misregistration rate comparison of images with VGA resolution

blocks of the designated size. The number of blocks in this image is then reduced to approximately 100 because each block is only one percent of the image size. If the resolution of two images is $m \times n$, the number of all eligible block pairs for the RIL block-pair search is consequently decreased to $100mn$. This is exponentially less than the $m^2 \times n^2$ of an exhaustive search. Therefore, the RIL block-pair search is accelerated by applying IBS. Figure 5.4 is the pseudo-code of IBS.

5.2.3 Richer Information and Likelier (RIL) block pair

The objective of creating the concept of RIL block-pair is to fix potentially increased image misregistration caused by a lack of complete information. The explanation in Subsection 4.4.4 addresses a new variation of the SAD function. This function is used to estimate the volume of identifying information and the degree of likeness of two

```
/* set the block size to be 1 percent of the image size */
blk_height = ceil(img_height/10);
blk_width  = ceil(img_width/10);
           ...
/* start IBS  */
for (each block of image 1)
{
    /* calculate self_sad */
    self_sad(img1, position of the block);

    /* find the likeliest block with minimum itb_sad in image 2 */
    min_itb_sad(img1, img2, position of the block);

    search_RIL_blockpair;
}
```

Figure 5.4: Pseudo-code of IBS

blocks. This variation of SAD involves MIN_ITB_SAD, Self_SAD and RIL_Rate. MIN_ITB_SAD represents the minimal intensity difference between blocks contained in two images, while Self_SAD illustrates the intensity difference between each pixel as compared to the average intensity of the block. RIL_Rate denotes the rate of Self_SAD over MIN_ITB_SAD. MIN_ITB_SAD and Self_SAD are computed in accordance with each block of image k generated by IBS.

To procure MIN_ITB_SAD of each block in image k, the intensity differences (ITB_SAD) between each block in image k, and all eligible blocks in image $k+1$ must be collected in advance. For each possible block-pair, the intensity of pixels located in the same position in both blocks is compared in pairs, and the differences between all pixel pairs are summed up to result in ITB_SAD for each block-pair. The smallest ITB_SAD is

set to MIN_ITB_SAD, and its associated block in image *k+1* is the likeliest match for the block in image *k*. The location of the block-pair is saved as well. However, in some cases, more than one pair of blocks have the same ITB_SAD which also result in MIN_ITB_SAD. In such cases, the first block-pair deemed to be MIN_ITB_SAD is considered as the likeliest block twin of image *k* and *k+1*. The position of this block-pair will not be replaced even if other block pairs with the same MIN_ITB_SAD are detected. Another problem is that MIN_ITB_SAD equals zero in some situations. This happens frequently at the coarse level because the data at this level only contains DC coefficients, which refer to the average values of 8×8 DCT units. Two blocks are likely to have the same intensity, and the intensity difference between them is zero. However, MIN_ITB_SAD cannot be zero for RIL_Rate calculation because it will be used as a denominator later on when the greatest RIL_Rate calculation is determined. In order to cater to the RIL_Rate calculation, PIMA discards block pairs for which MIN_ITB_SAD are zero, even though they might actually be alike. The block-pair with the smallest non-zero MIN_ITB_SAD could be slightly offset, but it is still reliable, because refinements can later correct any minor misregistration at the two increasing quality levels. Therefore, the accuracy of registration does not suffer by discarding block pairs with zero MIN_ITB_SAD.

The purpose of introducing Self_SAD is to obtain a pair of blocks which has as much significant information as possible. If a pair of blocks has a great Self_SAD with a small MIN_ITB_SAD, this will result in a block-pair with sufficiently distinguishing contents. When a pair of similar blocks has rich and varied information, the mismatch probability of such blocks is lower than that of two blocks that only have a minimum difference. Hence, fake block pairs are effectively filtered out. RIL_Rate, which divides Self_SAD by MIN_ITB_SAD, is used to identify a pair of rich-content blocks with which to achieve precise image registration, even though the information is not complete at the coarse level. Figure 5.5 shows the pseudo-code for the RIL block-pair calculation.

```
/* initialize the rate of RIL block-pair*/
coarse_ril_rate = 0;
      ...
/* In the loop of IBS  */
for (each block of image 1)
{
  self_sad(img1, position of the block);
  min_itb_sad(img1, img2, position of the block);

   /* discard block pairs with zero min_itb_sad */
  if (min_itb_sad != 0){
   if (self_sad/min_itb_sad) > coarse_ril_rate)
   {
       /* replace ril_rate with current blockpair's rate */
       coarse_ril_rate = self_sad/min_itb_sad;

       replace address with that of current block of image 1;
       replace address with that of current block of image 2;
   }
}
```

Figure 5.5: Pseudo-code of obtaining the RIL block-pair

PIMA registers images using segmental image information contained in the coarse level, and this segmental information is only common information in 8×8 blocks. The risk of misregistration occurred in PIMA is obviously higher than that occurred in traditional mosaicking approaches. The use of RIL block pair successfully reduces the number of misregistration to a minimal level, and limits the misalignment to slight displacement. This minor misalignment can be remedied by hierarchical refinement using a small range

search. As a result, the accuracy of PIMA's image registration is improved, and the efficiency of a surveillance application using PIMA is enhanced.

5.2.4 Small Range Search (SRS)

Correct image registration is a crucial requirement in order to generate an ideal and seamless synthesized image for a surveillance application user. Small Range Search is invoked for registration refinement at the two finer quality levels after the initial registration is accomplished at the coarse level. The RIL block-pair obtained during registering two images at coarse level has a small non-zero MIN_ITB_SAD, which means that existing information contained in each of the two blocks is not exactly the same. This probably because the two RIL blocks are slightly displaced. Therefore, refinement is necessary to correct this displacement, and can happen as more information is added at the higher-quality levels. In addition, the RIL block-pair efficiently registers the images at the highest level of accuracy at the coarse level because it contains abundantly distinct contents. Major registration errors have been eliminated during the initial alignment. Refinement can then occur within a bounded range of the neighbourhood.

For the sake of deciding on the search range, neighbours within 1, 2 and 3 steps in all directions are examined. Blocks beyond 3 steps are unnecessary because the wide-range shift will bring in lots of insignificant data or lose prominent information contained in the blocks, which is useless for refinement. Experiments have been conducted to determine a satisfactory search range. Table 5.3 shows the rates of successful refinement.

Neighbourhood	Number of Misregistration	Number of Successful Refinement
1-step	10	8
2-step	10	10
3-step	10	10

Table 5.3: Rates of successful refinement within different neighbourhood

Results point out that blocks within 1 step do not contain sufficiently salient infor-

mation to correct the misregistration, while neighbours 2 and 3 steps away succeed in accomplishing the refinement. To reduce the computational load, a 2-step neighbourhood is chosen as the suitable range to search a pair of best-matched blocks. In other words, 25 candidates are found to be eligible for the RIL block in each of the two images. At each of the two higher quality levels, 625 pairs of eligible blocks are compared, so that a gradual refinement of image registration can be achieved.

Within the defined small range, an exhaustive search is employed to match each pair of possible blocks. ITB_SAD is the only metric required for similarity measurement for registration refinement. At each of the two higher quality levels, the block pair with a minimum ITB_SAD is updated to become the RIL block pair, and its corresponding location in both images is used for image stitching. Figure 5.6 shows the pseudo-code of the Small Range Search.

5.3 Summary

The procedure of PIMA's implementation has been addressed in depth in this chapter. PIMA's workflow has been introduced step-by-step, image registration, the cornerstone of PIMA, has been stressed in accordance with its four specific features. For the purpose of adapting PIMA to a visual surveillance application, concerns with regard to the accuracy and efficiency of PIMA have been discussed, coupled with their corresponding solutions. In a word, PIMA is designed to significantly improve the efficiency of a visual surveillance application using a WISN.

```
for (each level)
{
   calculate new startpoint for RIL_block within 2-step
neighbourhood of image 1;
   reset RIL_block startpoint in image 1;

   calculate new startpoint for RIL_block within 2-step
neighbourhood of image 2;
   reset RIL_block startpoint in image 2;

   for (each block of image 1)
   {
      for (each block of image 2)
      {
         itb_sad(img1, img2, block1, block2);

         if (itb_sad < min_itb_sad)
         {
             min_itb_sad = itb_sad;

             replace address with that of current block of image 1;
             replace address with that of current block of image 2;
         }
      }
   }
}
```

Figure 5.6: Pseudo-code of SRS

Chapter 6

Experiments and Results Analysis

In order to evaluate PIMA's performance and validate its efficiency, experiments are designed and conducted after introducing the implementation of PIMA in Chapter 5. The output of PIMA is compared to that of traditional techniques. Analysis of the experimental results proves that PIMA achieves precise image registration, and provides a novel display mode with notable quality. In particular, PIMA effectively shortens the current delay that occurs before the first view. In brief, the efficiency of a visual surveillance application can be significantly improved by using PIMA.

6.1 Experiments Design

The intention of designing PIMA is to enlarge Field-Of-View (FOV) of a realistic view without a noticeable delay, so as to enhance the performance of a visual surveillance application using a WISN. Thus, experiments should be carried out in a WISN environment, where source images are collected and transmitted. However, a lack of the equipments such as wireless camera sensors meant that experiments could not be conducted in a real environment. Network Simulator 2 (NS-2) [30] was used to create a WISN, and simulate the image transmission over it. In order to simulate image acquisition, various images were stored in a hard drive in advance. They were then read into

buffers and transferred at the specified point in time to simulate the process of images being sent by sensors. PIMA was executed at the sink node of the WISN when it is receiving the images. Because Scans Preference of PIMA is integrated in RSTP, so that RSTP is the designated image transport protocol for PIMA used in NS-2. In addition, multiple groups of images, each featuring different resolutions and content, are used for mosaicking. Because PIMA mosaics images using incomplete information to generate an early preview so as to shorten a user's waiting time, therefore, experiments were designed according to delay, display and registration accuracy and are described as follows.

6.1.1 Delay Measurement

The delay before an image's first display is caused by considerable processing time involved in image transmission and image mosaicking. Two methods can be used to assess the delay. Firstly, the size of a file is strongly associated with the transmission speed of this file under the same limited conditions of wireless links, thus, the size of the data required for the first display can be used to imply the processing time of data transmission. In other words, the first scan sizes of Progressive JPEG (P-JPEG) files and Baseline JPEG (B-JPEG) files are compared to reflect the transmission speed. Secondly, the overall processing time including image transmission and image mosaicking can also be collected and compared to measure the delay. Since Scans Preference of PIMA has been added to RSTP, RSTP can be used for transmitting P-JPEG images for PIMA. By comparison, Transmission Control Protocol (TCP) is invoked to transfer B-JPEG images. Figure 6.1 shows the topology of a WISN used for image transmission in NS-2. Images are sent from two senders through wireless connections and received by the sink, at which point mosaicking algorithms are executed. The bandwidth of the wireless connection is set to 11Mbps, and the package loss rate is changed accordingly for comparison with the transmission speed. Images used for simulations have a 352×288 resolution, and are clustered into two groups. One group contains P-JPEG files used by RSTP, while the other group contains B-JPEG files used by TCP.

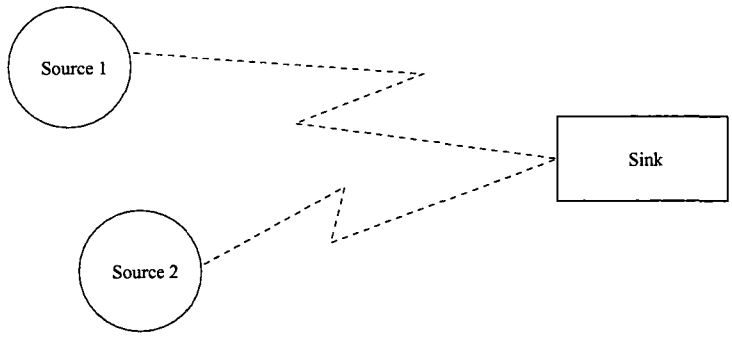


Figure 6.1: Simulation topology for image transmission in a WISN

Table 6.1 shows parameter values for the simulation of image transmission and mosaicking. Loss rates of wireless connections will be set to 0%, 10%, 20% and 30%, respectively, in order to compare the overall processing duration between PIMA and traditional approaches.

Parameter	Value
Image Resolution	352×288
Image Compression Standard	JPEG
Image Size	~ 15KBytes
Number of Sensors	2
Bandwidth	11Mbps
MAC Layer Protocol	IEEE 802.11b
Queuing Policy	DropTail
Interface Queue Length	1000 packets

Table 6.1: Simulation parameters of image transmission

An exhaustive search for the likeliest block-pair is impractical if an optimal performance evaluation is to be achieved. To estimate the processing time of mosaicking as compared to that of PIMA, Image Blocking Search (IBS) is used in a traditional algorithm to seek a likeliest block-pair. In summary, overall delay comparisons have been made between RSTP-PIMA and TCP-IBS at different loss rates of wireless links.

6.1.2 Display of the Mosaicked Image

The amount of the information a viewer obtains in one glance is one of the important aspects for evaluating the efficiency of a surveillance application. In other words, the amount of recognizable information contained in every view or display is significant for a surveillance application's performance. The mode and the quality of a display are two phases to access the volume of recognizable details in each display. The display mode is the way a surveillance application displays the mosaicked image. An effective display mode should provide a viewer as much information as possible in a display. The display quality is the definition of a display, and can be referred to the quality of the displayed image. It is no doubt that a clearer image has better quality for display.

For the purpose of validating PIMA's effect on improving the performance of a surveillance application in a WISN, PIMA's output is compared to the output produced by traditional techniques according to the display mode and the display quality. The display mode includes the way an image is displayed and the Field-Of-View (FOV) of this image. Peak-to-peak signal-to-noise ratio (PSNR) [23] is a popular criterion in order to measure the quality of an image reconstructed by a lossy compression codec.

As stated in Chapter 2, certain data is lost when an image is compressed. The decompression codec introduces noises to the original data when it reconstructs this image. Although there is no visible difference between the original image and its decompressed version, the data contained by the reconstructed image is not exactly the same as the data in the original one. Therefore, PSNR is used to indicate the noise ratio in the image data compared to the original data. Given the size of an image is $m \times n$, Equation 6.1 [23] is a mathematical definition of PSNR. In general, a decompressed image with a greater PSNR value indicates its reconstruction is of higher quality.

$$d(X,Y) = 10\log_{10}\frac{255^2 nm}{\sum_{i=1,j=1}^{n,m}(X_{i,j}-Y_{i,j})^2} \tag{6.1}$$

MSU Video Quality Measurement Tool [23] is a common tool for PSNR calculation, and is used to computer PSNR of PIMA's output and the output generated by traditional algorithms. Since only grayscale images are used for PIMA's implementation and experiments, The PSNR of luminance component Y (Y-PSNR) is chosen to be the metric representing PSNR for display quality measurement.

6.1.3 Registration Accuracy

In order to assess the registration accuracy of PIMA, PIMA's output was compared to the output of prevalent registration techniques. SAD, SSD and NCC, as stated in Subsubsection 3.3.1.1, are the common similarity metrics for patch-based registration algorithms. The RIL block-pair developed for PIMA is a variation of the SAD function. Therefore, the final registration output of RIL used by PIMA (PIMA-RIL) was compared to the output of SAD used by IBS, SSD used by IBS, and NCC done by Matrix Laboratory (Matlab). In addition, registration accuracy also lies in registering images that have different contents. Thus, images taken from different environments have been registered by PIMA for accuracy assessment.

6.2 Results Analysis

The performance of PIMA is evaluated by analyzing the experimental results with respect to delay measurement and accuracy of registration, which were designed for experiment conduction. Moreover, comparisons according to display mode and algorithm complexity were made so as to evaluate PIMA's efficiency.

6.2.1 Delay Measurement

Delay, reflecting an application user's waiting time, is the cumulative processing time of image transmission and image mosaicking. The first scan size of the image serves as

the reference to estimate the delay that will occur before the initial display of the mosaicked image. Reliable measurement can be performed by simulating image transmission followed by image mosaicking.

6.2.1.1 Size comparison of the first scan

Traditional image mosaicking algorithms do not begin the mosaicking process until images data is completely received. However, PIMA is able to begin to mosaic as soon as the first scans of different images are received. If multiple images are sent simultaneously through wireless connections under same conditions, the first scan size of the image can reasonably predict the delay involved the transmission. In essence, a smaller scan size means a decreased transmission duration and a shortened delay. In order to compare the size of the first scan, several images of different resolutions are encoded into both P-JPEG and B-JPEG formats. The image size of P-JPEG is slightly larger than, or roughly equal to, the size of B-JPEG. Thus, the total time for transmitting an entire P-JPEG file approximates the time for transmitting a B-JPEG file. A B-JPEG image is stored as one scan, and the first scan of this image is the whole file. A P-JPEG image is stored in multiple scans, and to obtain a comparable size, the first scan of this image is decoded and re-encoded into a B-JPEG file. In order to demonstrate the difference of the first scan size between a B-JPEG file and a P-JPEG file, one hundred various images are divided into five groups according to difference resolutions. Twenty images in each group are encoded into both B-JPEG and P-JPEG format. The first scan of every P-JPEG image is separated from the complete P-JPEG file so as to be compared to the B-JPEG file. Table 6.2 and Figure 6.2 illustrate the difference between B-JPEG and P-JPEG with regards to the first scan size. In order to validate experiments for every group of images, an error bar, which is shown in Figure 6.2, are used to represent 95% Confidence intervals (95% CI) of experimental results.

The results in Table 6.2 show that the first scan size of a P-JPEG image is from 21% to 30% of the first scan size of a B-JPEG file. Thus, sending the first scan of a P-JPEG

image is faster than sending the single B-JPEG scan. In other words, within the same period of image transmission, PIMA is able to complete the image mosaicking process and deliver the first approximate scene much earlier than in the traditional method.

Resolution	1st scan size of B-JPEG	1st scan size of P-JPEG	%
325×288	15KByte	5KByte	30
486×341	30KByte	8KByte	26.67
640×480	81KByte	20KByte	24.69
817×700	112KByte	25Byte	22.32
1221×818	206KByte	44KByte	21.36

Table 6.2: Comparison table of 1st scan size of B-JPEG and P-JPEG

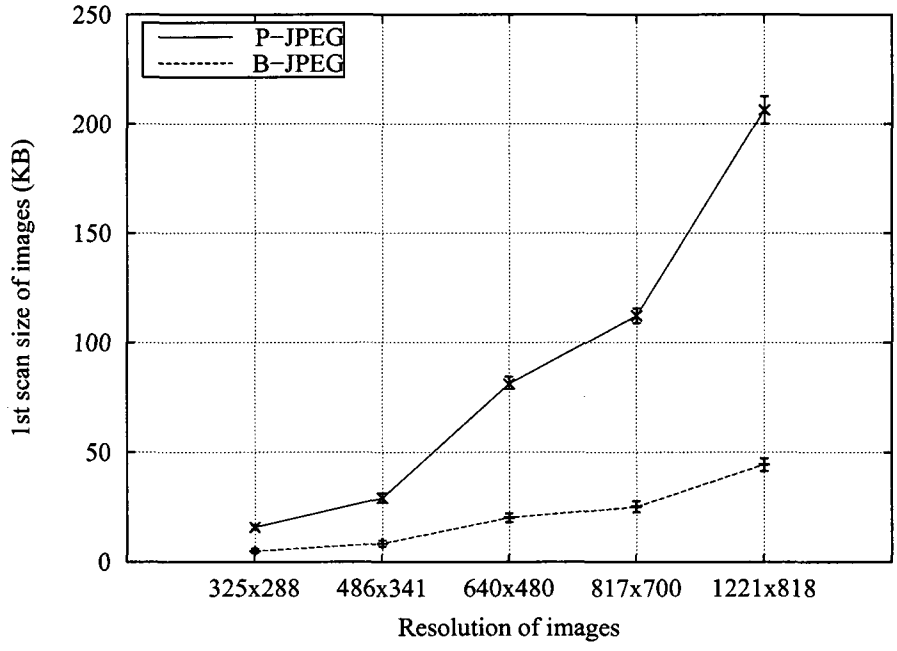


Figure 6.2: Comparison graph of 1st scan size of B-JPEG and P-JPEG

Results in Figure 6.2 indicate that the percentage of the first scan size over the entire file decreases when the resolution of the image is higher. This means that the size of the first scan is much smaller than that of the whole file in the case of a high-resolution image, and thus, the transmission of the first scan is much faster than the transmission of the whole file. Therefore, by applying PIMA in a WISN using high resolution images, the delay that occurs before the first display can be significantly shortened. PIMA proves that it can support high-resolution image mosaicking over WISNs.

6.2.1.2 Delay comparison before the first display

Simulations were carried out using NS-2 to estimate the cumulative delay of mosaicking images received from a WISN. The real-time scheduler of NS-2 is designed to synchronize the execution of a process with real-time, and is used to obtain the transmission time of scans or images. Two groups of overlapping images mentioned in Section 6.1 were used for the simulations, while P-JPEG files were used by RSTP-PIMA and B-JPEG files were used by TCP-IBS. The package loss rate of error-prone wireless connections was set to 0%, 10 %, 20% and 30%. The simulation for every loss rate has been run fifty times. Figure 6.3 shows the display time points of three scans for coarse-to-fine display by RSTP-PIMA, and the display time points of TCP-IBS at various loss rates. Error bars of 95% Confidence Interval (95% CI) are also shown in Figure 6.3.

Results in Figure 6.3 show that the display time for fine quality level of RSTP-PIMA is almost the same as the display time of TCP-IBS, which means, the complete processing time of RSTP-PIMA is roughly equivalent to the processing time of TCP-IBS. However, RSTP-PIMA is able to offer two early previews before the final display. The first preview is usually generated within one second even the loss rate of wireless connections is as high as 30%. PIMA outperforms traditional approaches by offering an early preview to a user of a surveillance application.

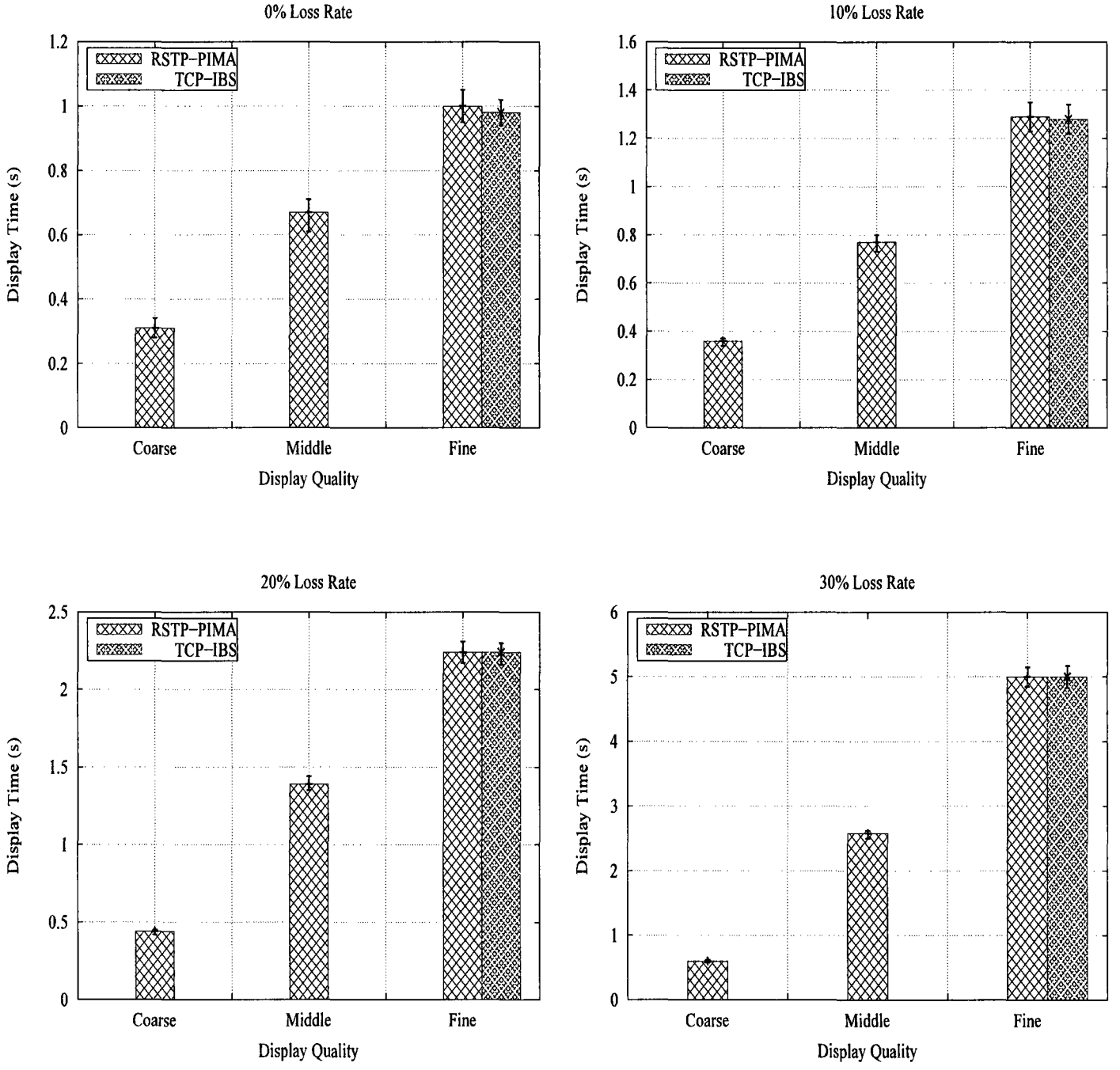


Figure 6.3: Time for every display of RSTP-PIMA and TCP-IBS at various loss rate

Since PIMA offers two previews before the final display, we considered the delay is the processing time of image transmission and the mosaicking process for the first preview. The delay measurement is taken from the point at which the first scan or image is sent, to the point when the first preview is displayed. A hundred group of images are used for the aforementioned simulation of image transmission and image mosaicking. RSTP-PIMA uses images in P-JPEG format, while TCP-IBS uses B-JPEG images. The average time of the delay before the first preview is calculated based on the data collected from the hundred experiments, and is used to represent the general duration before the first display. Figure 6.4 shows the average time of the delay before the first display of RSTP-PIMA and TCP-IBS.

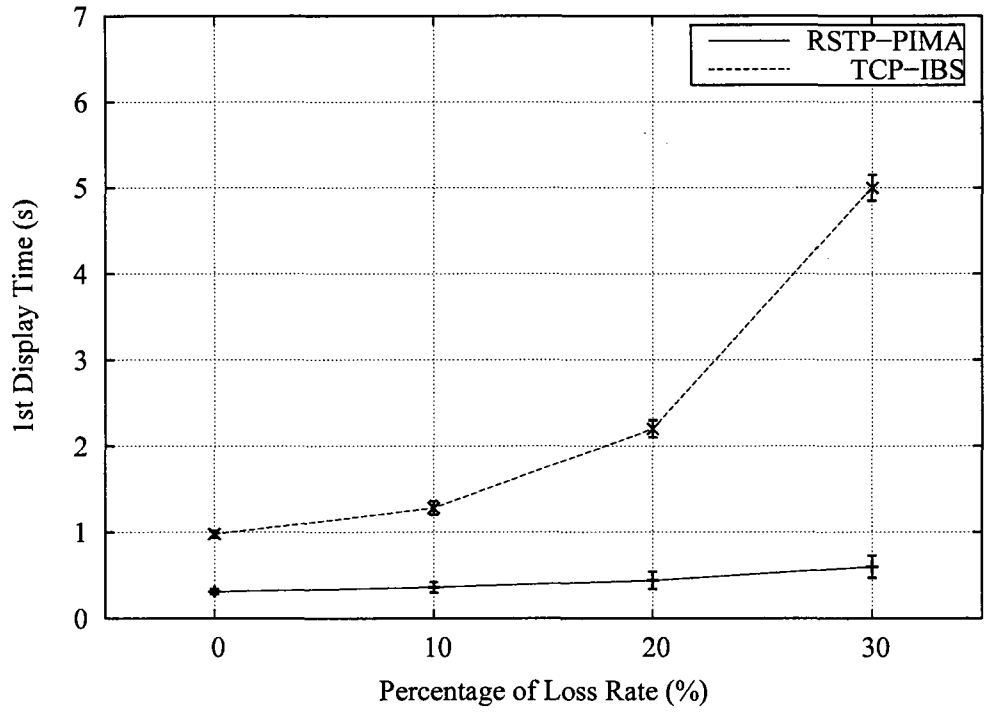


Figure 6.4: Comparison of the delay before 1st display of RSTP-PIMA and TCP-IBS

Figure 6.4 shows that PIMA is able to generate the first display of a synthesized image less than one second even if the package loss rate is 30%. In contrast, delay before the display generated by the traditional algorithm TCP-IBS is from one second to five seconds when the package loss rate is increased from zero to 30%. Simulation results indicate that PIMA displays the rough preview of a merged image significantly earlier than does a traditional approach using IBS. This traditional approach does not mosaic images until it receives all image data. Nevertheless, PIMA starts processing as soon as the first scans are received. At the same time, the sink node is still receiving other scans for the next mosaicking. Therefore, PIMA successfully reduces the application user's waiting time during the first view. Moreover, results also show that the delay is significantly decreased by using PIMA in cases where the package loss rate is high. PIMA is suitable to be used in a unreliable WISN that data loss is occurred frequently.

In conclusion, PIMA is proven to effectively shorten a user's waiting time before the initial display of a fully synthesized image. Therefore, it is best suited to mosaic high-resolution images received from unreliable wireless links in a WISN.

6.2.2 Display of the Mosaicked Image

An efficient surveillance application should offer a recognizable view that contains rich details of a monitored area in a short time. Thus, a viewer of this application can obtain enough information in one glance without waiting for the next display. PIMA uses image mosaicking to achieve a large Field-Of-View (FOV), and offers a new display mode that the quality is coarse-to-fine with a recognizable and early preview. This enlarged FOV and progressive-quality display mode allows a user to get more information from fewer displays of a interested region within a shorter processing period, so as to improve the efficiency of a surveillance application in a WISN.

6.2.2.1 Display mode

Most of the current visual surveillance applications render and display images using the same images they receive. Information contained in each image is limited because the FOV of a sensor camera is small. More views are required so as to display the details of the entire monitored area. In order to improve the efficiency of such a surveillance application, PIMA invokes techniques of image mosaicking before it renders a view. PIMA stitches every two images together to achieve a large FOV, so that more information is contained in each view PIMA renders. In other words, the data in every PIMA's view is the data contained in two views of traditional approaches. This result in an enhanced performance of a surveillance application, of which a user can obtain more information using fewer views.

In addition to a magnified FOV, PIMA displays each view in a manner of coarse-to-fine. Although existing approaches mosaic images in different ways, all of them require reference images available as a whole, and display their products from top to bottom. However, the progressive quality of the image display is a novel feature of PIMA enabled by the presence of P-JPEG. P-JPEG is outstanding due to its ability to exhibit a progressive quality of display, as the image file is split into a number of scans and displayed in order. PIMA makes use of this important feature of P-JPEG in order to split the display into three stages so that users have a first glance of the image without having to wait a long time. The image quality is continuously refined later on until the final scene is produced. Figure 6.5 illustrates a coarse-to-fine display with an enlarged FOV generated by PIMA and a top-to-bottom display for each view by a traditional approach.

Moreover, the coarse-to-fine display provides the user a choice: either wait for the refined image views or skip them if the current view provides sufficient information about the image. In some circumstances, such as, when a user is looking for a specific object, compared to either a top-to-bottom or a low-to-high resolution display, the coarse-to-fine display is superior to both because the user can seek out the anticipated detail within the rough view early on, and switch to a view of another area if it is not found.

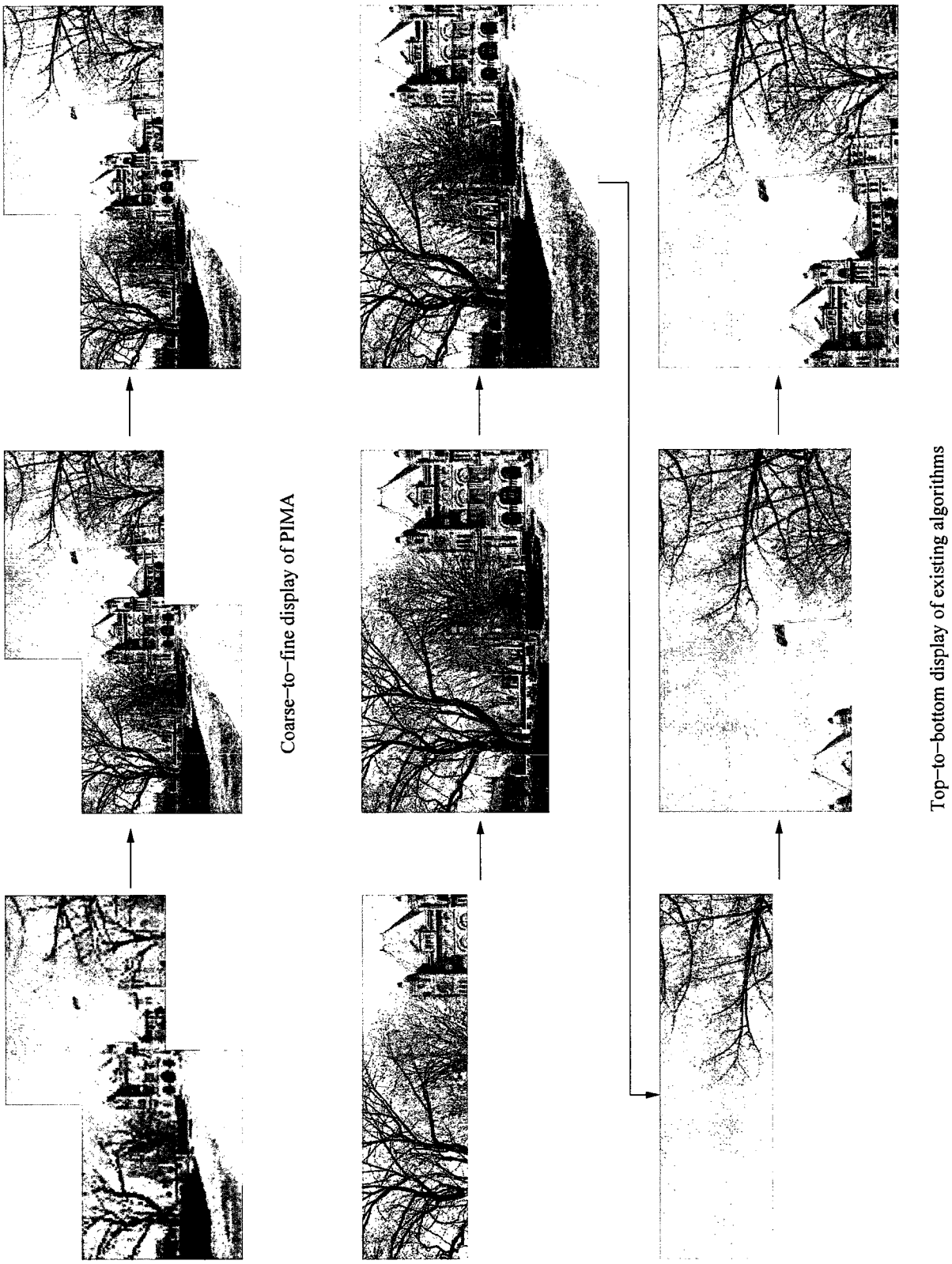


Figure 6.5: Comparison of display modes produced by PIMA and traditional approaches

PIMA's gradual-improvement display mode is appropriate for situations in which mosaicking is applied to images collected from a low-speed WISN. Since image mosaicking is computationally intensive and time-consuming, the delay before the integrated image is rendered can be extensive. Ideally, the final product of image mosaicking should be exhibited in a coarse-to-fine manner while the full scope of the image remains unchanged.

6.2.2.2 Display Quality

Prevailing surveillance applications and current image mosaicking approaches use Baseline JPEG (B-JPEG) as the default operation mode to compress images. B-JPEG images are encoded in one scan and displayed from top to bottom. PIMA inventively uses Progressive JPEG (P-JPEG), and encoding images into several scans. By invoking P-JPEG, PIMA is able to offer a user an early preview of a monitored area in order to shorten the user's waiting time. Moreover, PIMA displays every view in a manner of coarse-to-fine. This display mode exceeds other display modes because it allows a user to know what is happening in the monitored area at his first glance. Therefore, PIMA enhances the performance of a visual surveillance application.

Since the first preview is the most important characteristic of PIMA, its quality is significant for PIMA's performance, and is highly concerned by a viewer. The perfect quality of the preview requires more image data, and results in slow transmission over an error-prone WISN. On the other hand, the preview with a poor quality cannot provide enough information, and is meaningless for the viewer. PIMA's performance will be degraded if either of these situations occurs. As mentioned in Subsection 4.4.1, DC coefficient contains the most significant data of every DCT block in an image using little space. PIMA designates an image's first scan to contain only DC coefficients, and uses this first scan to produce a preview. As a result, this preview is generated much earlier than the final view, and is able to provide a viewer the most significant information of an image. In order to evaluate the quality of every display of PIMA, displays at three level of quality will be compared to each other. Furthermore, SAD, SSD and NCC, which

are three common registration methods using B-JPEG images for image mosaicking, represent the traditional mosaicking approaches. They are implemented so as to generate their top-to-bottom displays. Each of these displays is divided into three stages: *Top, Middle* and *Complete*, according to the display scope of an image. The quality of three display stages are compared to the quality of PIMA's three display levels: *Coarse, Middle* and *Fine*, respectively. Because grayscale images are used for implementation, peak-to-peak signal-to-noise ratio of brightness component Y (Y-PSNR) is assigned to be the error metric for the quality comparison. Higher Y-PSNR indicates better quality of an view or display, and vice versa. Figure 6.6, 6.7 and 6.8 illustrate display quality comparisons for three sample images.

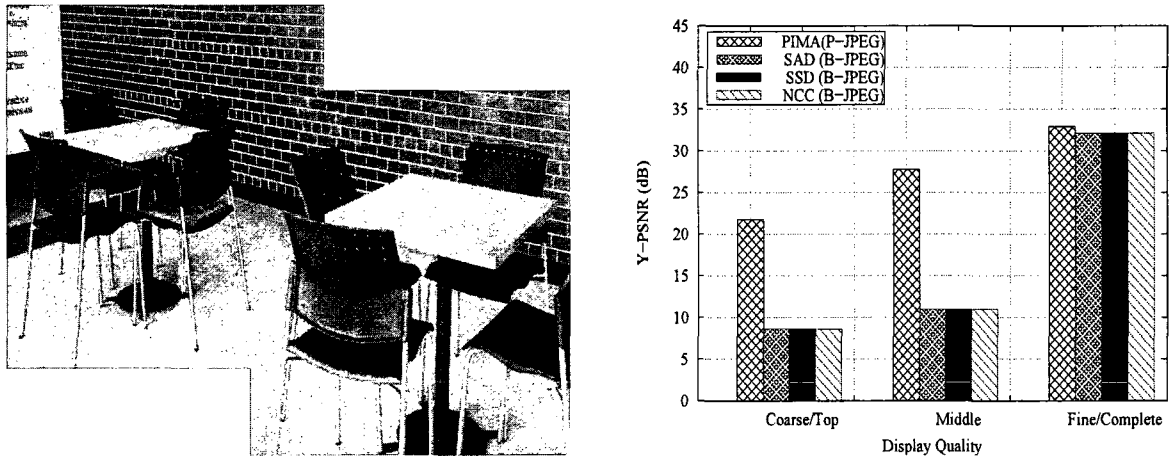


Figure 6.6: Y-PSNR of image taken for indoor scene under sunshine

Results in three figures show that views produced by SAD, SSD and NCC at each stage are of the same level of quality. This is because all of the common methods use B-JPEG images for mosaicking. Also, due to the same reason, views are displayed from top to bottom. Views displayed at the top and middle stages are parts of an merged image. Due to the lack of information contained in the missing part of an image, the quality of views at top and middle stages are not good enough. The user of a traditional surveillance application cannot get full comprehension of a monitored region from views at the first two stages. They have to wait until the complete view of an image is displayed. Results

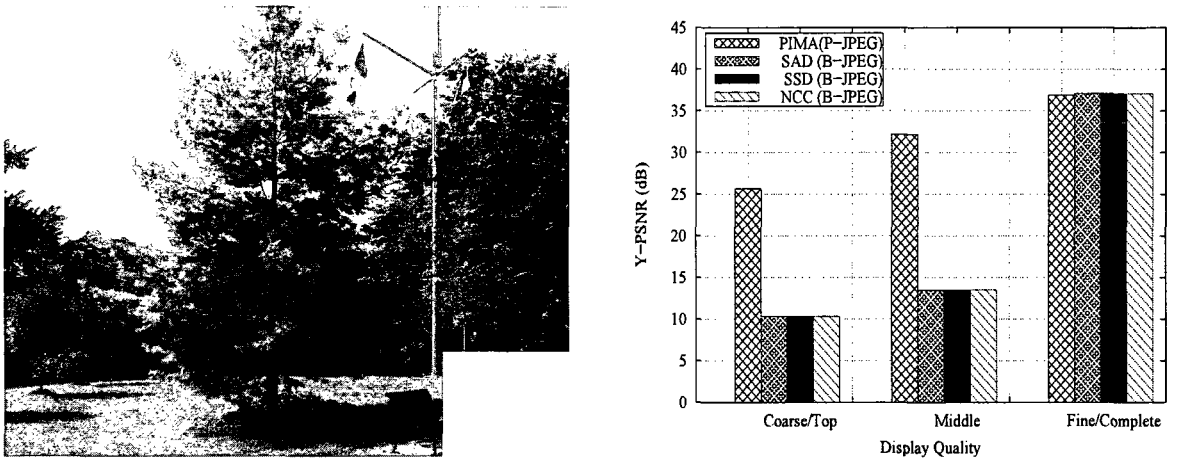


Figure 6.7: Y-PSNR of image taken for outdoor scene under sunshine

in above figures also show that PIMA can not only generate a compatible quality for the final display as compared to the three common methods, but also produce a preview at an acceptable quality. The quality of the first display is reasonably worse than the final display because only the most significant information of an image is presented. However, Y-PSNR's value of the first display is close to that of the final display of each sample image. In other words, the first preview is clear enough for a viewer to obtain his basic knowledge about an image. In addition, the quality of the first preview produced by PIMA is much better than the quality of the display at the top stage of three common methods. Experimental results demonstrate that PIMA is able to offer an early and understandable preview at a desirable quality in comparison to the display at the top stage of current approaches.

In summary, PIMA inherits multi-scan feature of P-JPEG, generates an early and approximate preview of each synthesized image, and displays each image with progressive quality. The quality of each display is ideal for a viewer in order to obtain the information about his interested area. By offering a new display mode with a satisfied quality, PIMA surpasses existing mosaicking approaches in enhancing the performance of a visual surveillance application in a unreliable WISN.

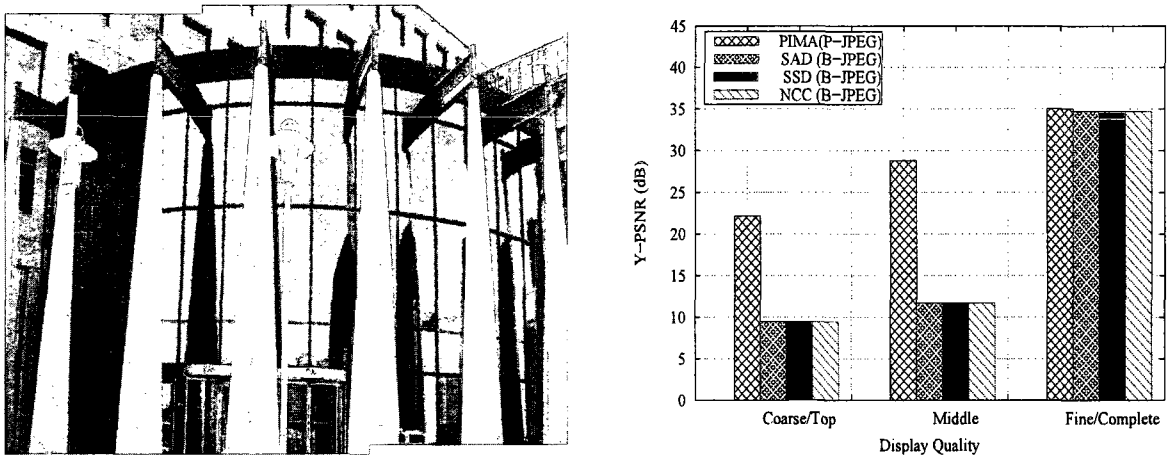


Figure 6.8: Y-PSNR of image having repeated patterns

6.2.3 Algorithm Complexity

Image registration is the most important process among all steps of image mosaicking, and generates the most computational overhead. The complexity of registration represents the complexity of the entire mosaicking algorithm. In the process of image registration, searching for the most alike block-pair is the step that incurs huge computational overhead. An exhaustive search is impractical because it seeks the most eligible block-pair in all possible positions in both images. If the resolution of the images is m×n, the complexity of an exhaustive search is $O(m^2n^2)$. Hierarchical motion estimation diminishes the computational load by downsampling original images twice to lower-resolution versions, and then seeking the likeliest block-pair from within the lowest resolution images. However, key identifying information could be blurred away during this downsampling process. An exhaustive search at the lowest resolution is always performed to improve the accuracy of registration. If the downsampling factor is 2, the complexity of hierarchical motion estimation is $O(\frac{m^2n^2}{256})$. Contrastingly, PIMA uses Image Blocking Search (IBS) to reduce the block number of one image to about 100, and develops the concept of RIL block-pair to ensure precise registration. Therefore, the number of eligible block-pairs in the search is limited to 100mn. The complexity of

IBS used by PIMA is then *O(100mn)*. PIMA is proven to be faster than an exhaustive search in general, and could be faster than hierarchical motion estimation if the image resolution is high.

It takes a comparable computational load to decode each scan of a P-JPEG file into a bitmap image. Thus, decompressing three scans of a P-JPEG file produces triple the overhead than decompressing a B-JPEG file. However, PIMA makes the assumption that the decoding process is conducted by a fast processor, so that the overhead becomes less of an issue. In addition, if application users choose to skip one or two of the later image displays, any subsequent scan(s) will not be decompressed. The computational load is then further decreased.

In summary, PIMA successfully preserves the algorithm complexity at a satisfactory level, and provides a coarse-to-fine display to shorten the delay before the image preview.

6.2.4 Output Comparison for Registration Accuracy

Prevalent algorithms that process image mosaicking rely on complete information, while PIMA proceeds with image registration based on fractional data so as to shorten delay. PIMA employs RIL block-pair to effectively reduce the number of opportunities for misregistration, and it aims to produce registration results equivalent to those of commonly used algorithms. To evaluate the registration accuracy of PIMA, PIMA's output is compared to the results generated by certain traditional approaches. Additionally, in order to assess the impact of a physical environment on registration accuracy of PIMA, PIMA's mosaicking output using images taken from various environments are presented.

6.2.4.1 Among RIL and existing similarity metrics

Traditional approaches taken to register images are based on the complete set of data of whole images. In particular, the similarity measurement of registration counts on having the correct data in order to minimize the risk of misregistration. Various similarity criteria, such as SAD, SSD and NCC, are used to estimate likeness with high precision. PIMA

registers images at the first scan, which contains only DC coefficients of DCT blocks. Although DC coefficients represent the most important information of DCT blocks, they only offer partial information about the whole image. This elevates the chance of misregistration taking place. In order to solve this problem, PIMA seeks a RIL block-pair, or a pair of alike blocks carrying richer information. The risk of misregistration is quite low if the block twins prominently feature sufficient information. Furthermore, PIMA hierarchically refines the initial registration at two subsequent quality levels, so that minor misregistrations are corrected before the final view is displayed. Figure 6.9 shows comparisons of registration results among the RIL block-pair used by PIMA and, SAD, SSD, and NCC.

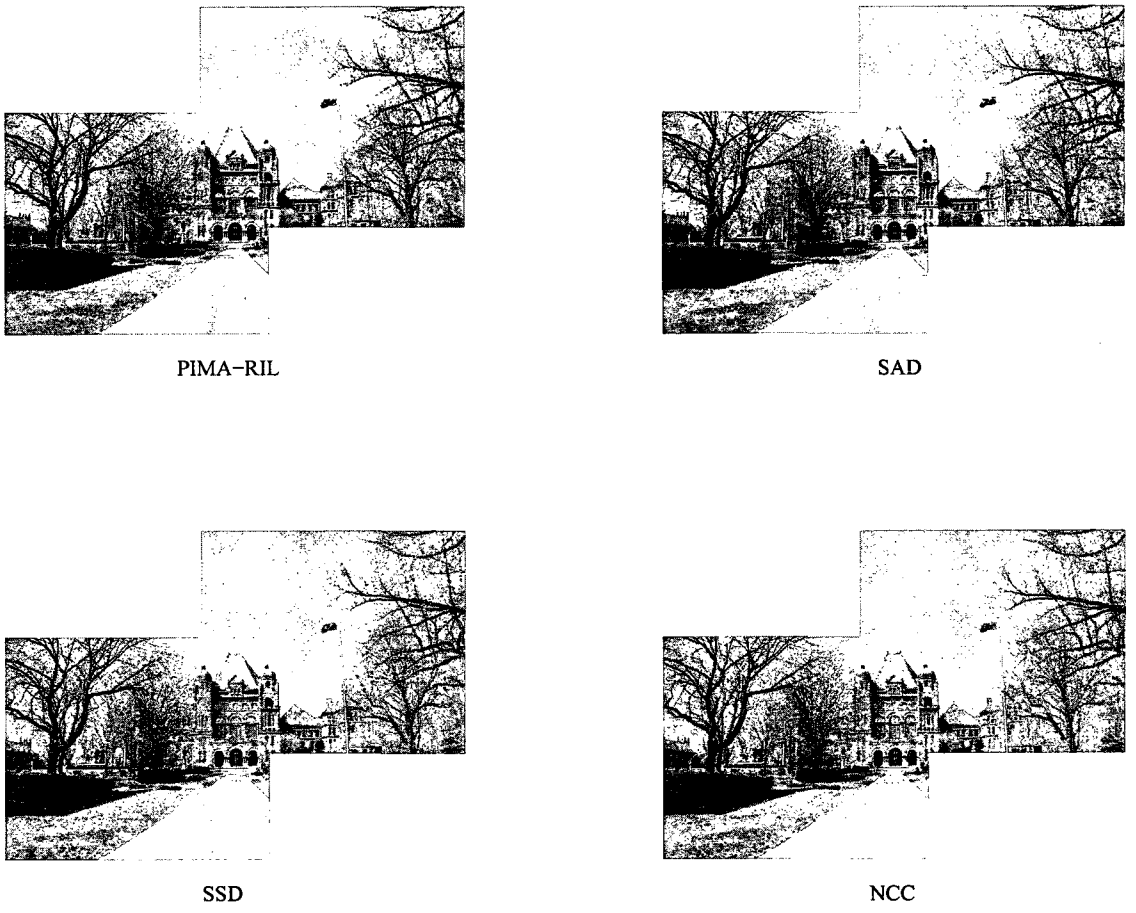


Figure 6.9: Final Output comparison for PIMA , SAD, SSD, and NCC

Figure 6.9 illustrates that PIMA is able to register images as precisely as the prevalent approaches even though it uses incomplete image data. In conclusion, the RIL block-pair of PIMA successfully filters out the feigned block pairs that do contain similar information but are not actually a match, and the hierarchical refinement gradually improves the accuracy of registration. As a result, PIMA produces the compatible wide-angle image compared to those of other popular algorithms.

6.2.4.2 Among images taken from different environments

In order to further evaluate the registration accuracy of PIMA and investigate PIMA's applicable environment, a variety of images with different contents were mosaicked by PIMA. The distribution and the volume of distinguishing information of these images were varied since the images were taken from different environments under different conditions. Scenes of the images include urban and rural areas, indoor and outdoor regions under sunshine and light, respectively. Moreover, repeated pattern is a tremendous problem for image registration because it is hard to identify and cause misrecognition. Either patch-based or feature-based algorithms have not completely solved this problem. For the purpose of investigate the impact of repeated patterns on accuracy of PIMA's registration, two groups of overlapping images with many repeated patterns were mosaicked by PIMA. Results indicate that PIMA is practical and reliable because it correctly registers images taken in a wide range of situations, including images with repeated patterns. Therefore, PIMA can be flexibly applied without any restriction on environment.

6.3 Summary

To properly evaluate PIMA's performance, a series of experiments have been carried out. Simulations were conducted with associated transport protocols in WISNs simulated by NS-2. Images with varying encoding modes, multiple resolutions or different contents were used. PIMA's performance was compared with that of existing algorithms or similarity metrics against the criteria of delay, display mode, complexity and registration accuracy. In summary, PIMA executes precise registration based on incomplete information of images, reduces delay by generating an approximate but recognizable image preview, provides a novel display mode with progressive quality, and retains the complexity at a satisfactory level. PIMA is ideal for obtaining an enlarged FOV for use in applications such as remote surveillance, in which source images are transmitted across error-prone, bandwidth-limited WISNs.

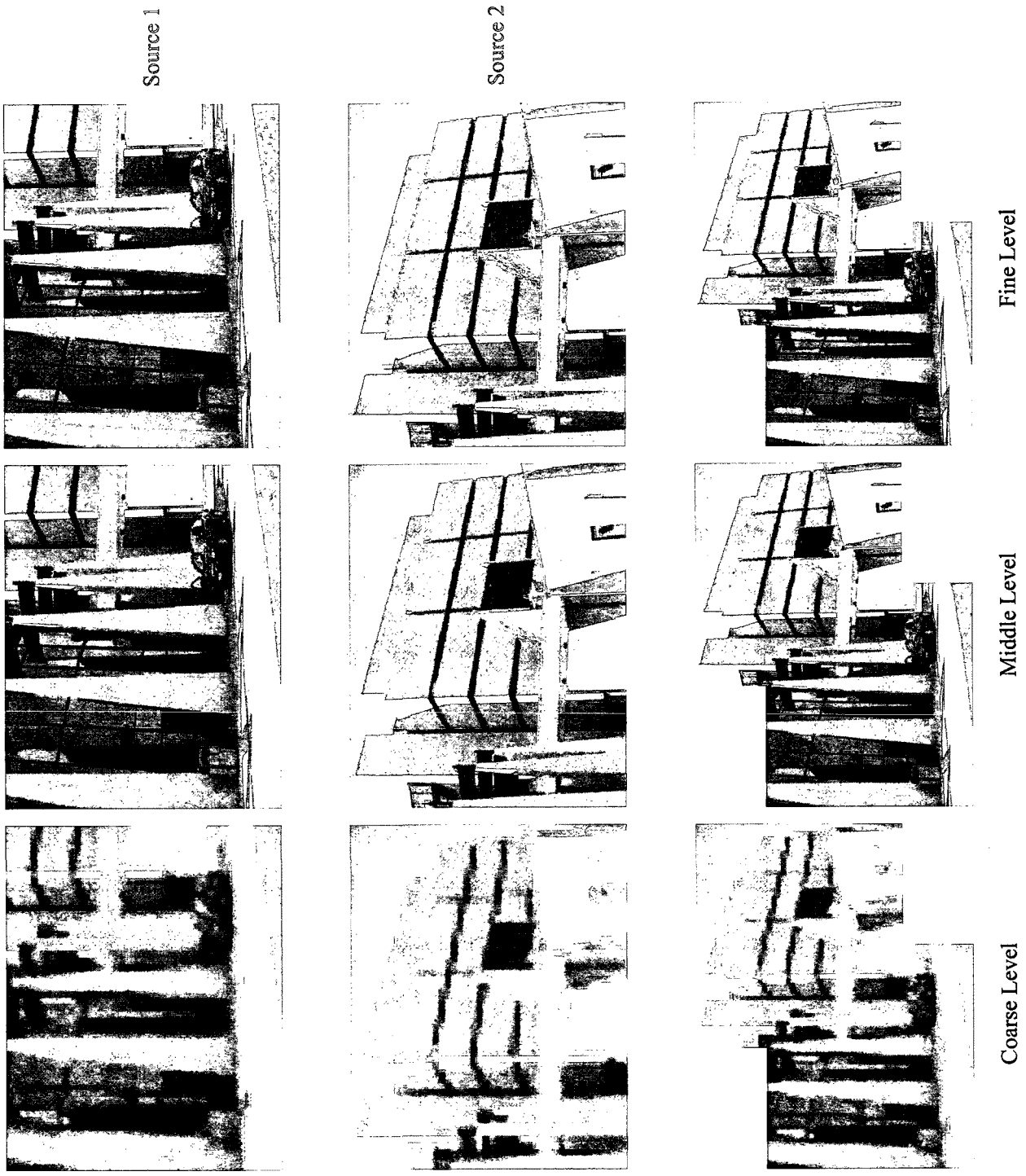


Figure 6.10: Progressive-quality display for images taken from urban region

Figure 6.11: Progressive-quality display for images taken from rural region

Figure 6.12: Progressive-quality display for images taken for indoor scene under sunshine

Figure 6.13: Progressive-quality display for images taken for indoor scene under light

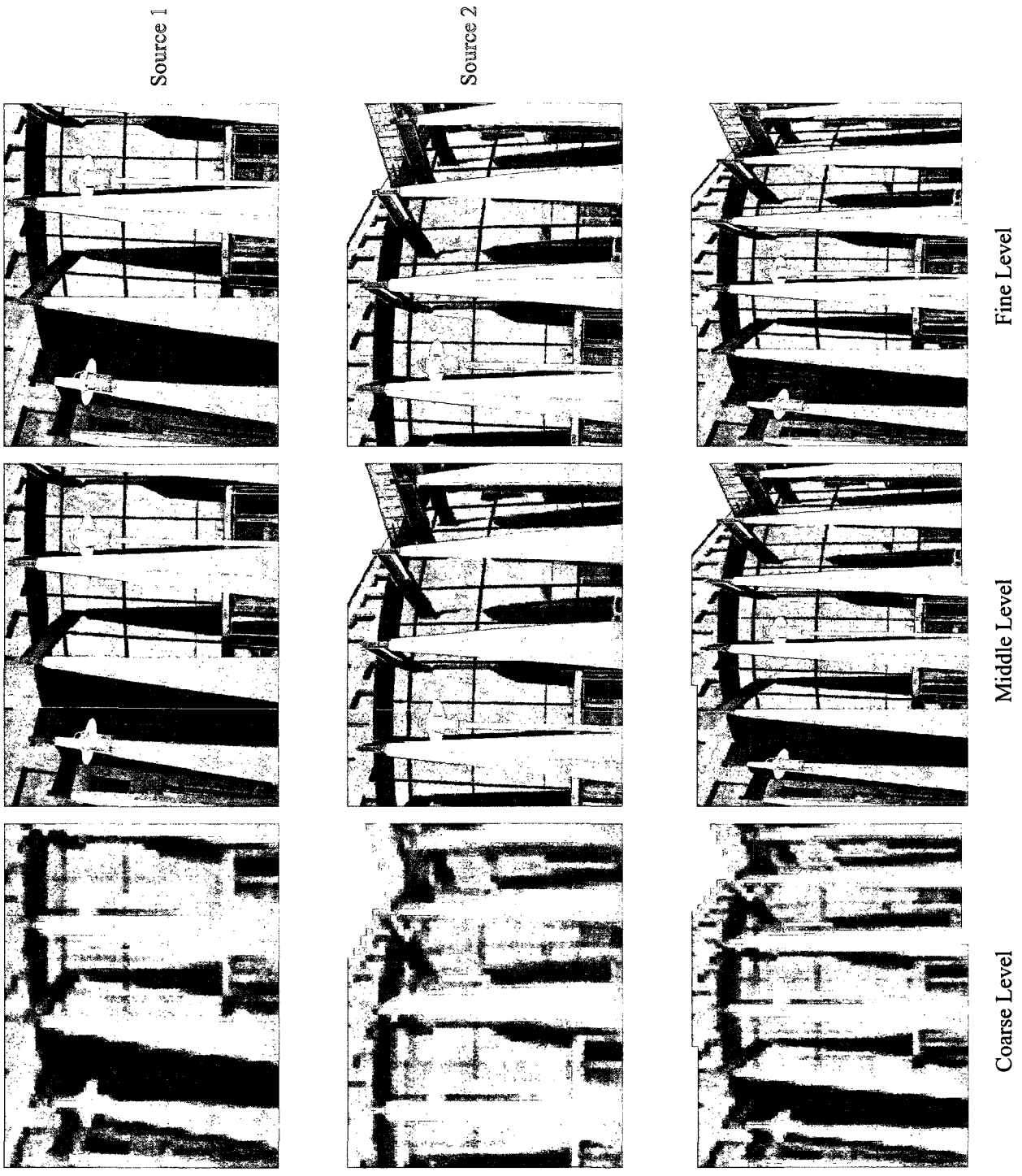


Figure 6.14: Progressive-quality display for images having repeated patterns

Chapter 7

Conclusion and Future Work

The deployment of a WISN in a visual surveillance application is limited because a noticeable delay caused by slow image transmission is observed. In addition, the efficiency of such an application is not high due to the small FOV of a single camera. Furthermore, the traditional top-to-bottom display does not allow a user to obtain the basic knowledge of a monitored region in a short time. In this thesis, a new image mosaicking algorithm, PIMA, has been proposed to overcome these problems. PIMA successfully produces the first approximate view of an image in a shorter time than the traditional algorithms, and offers a progressive quality display with an enlarged FOV. In this chapter, the thesis work will be reviewed and summarized, and possible issues for future study will be addressed.

7.1 Conclusion

In order to design PIMA, related work was introduced in the first two chapters. The JPEG still image compression standard was examined in Chapter 2, accompanied by an in-depth exploration of the relevant encoding and decoding processes. JPEG is currently the most prevailing compression standard and image format available because it can significantly reduce file size without degrading image quality. Moreover, JPEG offers a progressive operation mode, which compresses an image into multiple scans of

sophisticated quality. Progressive JPEG outperforms other JPEG encoding modes on transmitting images over low-speed WISNs because it can display a quick and coarse preview based on segmental image data, and gradually refine the image quality once additional data is received. This multi-scan attribute of Progressive JPEG is potentially beneficial for a visual surveillance application, specifically, in terms of shortening the delay before the first display of the synthesized image. Since image mosaicking is chosen to produce an image with an enlarge FOV for visual surveillance, the procedure for image mosaicking was introduced in Chapter 3. Algorithms associated with each image mosaicking step were explored. The cornerstone of image mosaicking is image registration, which deduces the degree of similarity between two images. Methods that enhance the accuracy and efficiency, and reduce the complexity of registration, were discussed. Algorithms were classified into two categories: patch-based and feature-based, and comparisons were carried out between these two approaches. Patch-based algorithms are found to suit surveillance applications that enlarge Field-Of-View (FOV) by mosaicking multiple images.

A noticeable delay is introduced during the mosaicking of images collected from error-prone and bandwidth-constricted WISNs. With the aim of shortening the amount of time users spend waiting for the merged image, as is the case in wireless surveillance, a new image mosaicking algorithm, PIMA, was discussed in detail in Chapter 4. The new algorithm aims to register images precisely using fractional data, assuming that Progressive JPEG is the preferable image format for image transmission over WISNs. PIMA applies mosaicking to three different quality levels, each decoded from different scans of the images, and displays a wide-angle view in a coarse-to-fine manner. PIMA's multi-display characteristic, inherited from Progressive JPEG, allows users to have a quick initial glance first and two increasingly refined views afterward. PIMA offers four essential features that enable the successful mosaicking of images using incomplete data. *Scan Preference* chooses three scans of which each contains significant information, to ensure correct registration, intelligible display and low computation of mosaicking. *Im-*

age Blocking Search accelerates the process of image registration by reducing the number of eligible block pairs. The concept of *Richer Information and Likeliest block pair* diminishes the potential misregistration caused by the use of partial data. *Small Range Search* amends registration within a bounded area in order to reduce computational overhead. These four factors gradually improve the accuracy of image registration while maintaining a manageable computational load. In brief, PIMA is able to display a recognizable preview with a broadened FOV during the reception of new image data, and the delay before first display is shortened. As a result, PIMA is adequately used in a visual surveillance application.

In Chapter 5, the implementation of PIMA was described in depth. Problems associated with the implementation were also addressed, along with corresponding experiments and solutions. In Chapter 6, experiments and results were presented according to various performance evaluation criteria. Results were extensively analyzed by comparing PIMA's performance with that of existing techniques. PIMA's image registration accuracy was presented by mosaicking images taken from a wide range of environments.

In conclusion, PIMA achieves precise image registration using incomplete image data, and shortens the delay before the first display of the image. In addition, PIMA displays the wide-angle views of a real scene with progressive quality, and maintains a high quality of performance as compared to existing patch-based algorithms. In particular, PIMA allows a viewer to obtain the basic knowledge of an interested region in a short time based on an early preview. By applying PIMA, wireless image sensor networks can be flexibly deployed in a visual surveillance application without any environment constraint, and the efficiency of the visual surveillance application can be eventually enhanced.

7.2 Future Work

Image mosaicking in a wireless image sensor network (WISN) is a challenging task due to the WISN's inherent limitation, such as finite resources and high error ratio. In order

to improve performance of applications using image mosaicking in a WISN, open issues are proposed for future development.

The resolution of images captured from different camera sensors cannot be uniform in certain applications because these applications need different configurations for camera sensors or various requirements. Research on mosaicking a collection of images of varying resolutions has become increasingly popular. In addition, the distributed registration of PIMA is derived from hierarchical motion estimation, which registers images using hierarchical levels of resolution. Therefore, PIMA can be enhanced to perform image mosaicking with multi-resolution images in the future.

Lastly, JPEG 2000 has a greater number of sophisticated features that support flexible image processing, such as progressive transmission, random code-stream access and processing and multiple resolution presentation. Techniques of Image-based Modeling and Rendering over WISNs can benefit significantly from such features. PIMA was created based on the progressive encoding of JPEG, therefore, it can be easily promoted to support JPEG 2000 in future applications.

Appendix A

Glossary of Terms

AC Alternating Current.

B-JPEG Baseline JPEG.

CC Cross-Correlation.

C_RIL Coarse level RIL.

DC Direct Current.

DCT Discrete Cosine Transform.

DoG Difference of Gaussian.

FOV Field-Of-View.

F_RIL Fine level RIL.

IBMR Image-based Modeling and Rendering.

ISO International Organization for Standardization.

ITB_SAD Inter-block_Sum of Absolute Difference.

JPEG Joint Photographic Experts Group.

M-JPEG Motion JPEG.

M_RIL Middle level RIL.

MIN_ITB_SAD Minimum inter-block SAD.

NCC Normalized Cross-Correlation.

PIMA Progressive Image Mosaicking Algorithm.

P-JPEG Progressive JPEG.

RIL Richer Information and Likelier block.

SAD Sum of Absolute Difference.

Self_SAD Self_Sum of Absolute Difference.

SRS Small Range Search.

SSD Sum of Squared Difference.

T_RIL Temporary RIL.

WISNs Wireless Image Sensor Networks.

WLANs Wireless Local Area Networks.

WSNs Wireless Sensor Networks.

Bibliography

[1] I. F. Akyildiz, T. Melodia, and K. R. Chowdhury. A survey on wireless multimedia sensor networks. *IEEE Wireless Communications*, 14(6):32–39, December 2007.

[2] Simon Baker and Iain Matthews. Lucas-kanade 20 years on: A unifying framework. *International Journal of Computer Vision*, 56:221–255, February-March 2004.

[3] A. Bartoli, M. Coquerelle, and P. Sturm. A framework for pencil-of-points structures-from-motion. In *Proceedings of Eighth European Conference on Computer Vision(ECCV 2004)*, pages 28–40, May 2004.

[4] J. R. Bergen, P. Anandan, K. J. Hanna, and R. Hingorani. Hierarchical model-based motion estimation. In *Proceedings of Second European Conference on Computer Vision (ECCV'92)*, pages 237–252, May 1992.

[5] M. J. Black and P. Anandan. The robust estimation of multiple motions: Parametric and piecewise-smooth flow fields. *Computer Vision and Image Understanding*, 63(1):75–104, January 1996.

[6] A. Boukerche, J. Feng, R. P. werner, Y. Du, and Y. Huang. Reconstructing the plenoptic function with wireless multimedia sensor networks. In *Proceedings of the 33rd IEEE Conference on Local Computer Networks (LCN 2008)*, pages 74–81, October 2008.

[7] M. Brown and D. Lowe. Recognizing panoramas. In *Proceedings of Ninth International Conference on Computer Vision*, pages 1218–1225, October 2003.

[8] P. J. Burt and E. H. Adelson. A multiresolution spline with applications to image mosaics. *ACM Transactions on Graphics*, 2(4):217–236, October 1983.

[9] C. Y. Chen and R. Klette. Image stitching - comparisons and new techniques. In *Proceedings of Computer Analysis of Images and Patterns*, pages 615–622, September 1999.

[10] Shenchang Eric Chen. Quicktime vr: an image-based approach to virtual environment navigation. In *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, pages 29–38, August 1995.

[11] A. M. Cheng and F. Shang. Priority-driven coding and transmission of progressive jpeg images for real-time applications. *Journal of VLSI Signal Processing Systems*, 47(2):169–182, 2007.

[12] O. Chum and J. Matas. Matching with prosac - progressive sample consensus. In *Proceedings of IEEE Society on Computer Vision and Pattern Recognition*, pages 220–226, June 2005.

[13] A. Agarwala et al. Interactive digital photomontage. *ACM Transactions on Graphics*, 23(3):292–300, August 2004.

[14] J. Matas et al. Robust wide baseline stereo from maximally stable extremal regions. *Journal of Image and Vision Computing*, 22(10):761–767, September 2004.

[15] S. Peleg et al. Mosaicing on adaptive manifolds. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 22(10):1144–1154, October 2000.

[16] V. Kwatra et al. Graphcut textures: Image and video synthesis using graph cuts. *ACM Transactions on Graphics*, 22(3):277–286, July 2003.

[17] Jing Feng. Image-based rendering on mobile devices. Master's thesis, University of Ottawa, September 2006.

[18] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, June 1981.

[19] J. D. Foley, A. van Dam, S. K. Feiner, and J. F. Hughes. *Computer Graphics: Principles and Practics*. Addison-Wesley, Reading, MA, USA, 1990.

[20] W. Forstner. A framework for low level feature extraction. In *Proceedings of Third European Conference on Computer Vision(ECCV'94)*, pages 383–394, May 1994.

[21] C.-S. Fuh and P. Maragos. Motion displacement estimation using an affine model for image matching. *Optical Engineering*, 30(7):881–887, July 1991.

[22] M. A. Gennert. Brightness-based stereo matching. In *Second International Conference on Computer Vision*, pages 139–143, December 1988.

[23] MSU Graphics and Media Lab. Msu video quality measurement tool. `http://compression.ru/video/quality_measure/video_measurement_tool_en.html`.

[24] N. Greene. Environment mapping and other applications of world projections. *IEEE Computer Graphics and Applications*, 6(6):21–27, November 1986.

[25] Frank R. Hampel, Elvezio M. Ronchetti, Peter J. Rousseeuw, and Werner A. Stahel. *Robust Statistics: The Approach Based on Influence Functions*. Wiley-Interscience, 2005.

[26] C. Harris and M. J. Stephens. A combined corner and edge detector. In *Proceedings Alvey Vision Conference*, pages 147–152, September 1988.

[27] R. I. Hartley and A. Zisserman. *Multiple View Geometry*. Cambridge University Press, Cambridge, UK, 2004.

[28] Zhengting He. Video compression and data flow for video surveillance. Technical Report CSD-91-646, Texas Instruments, September 2007.

[29] Peter J. Huber. *Robust Statistics.* Wiley-Interscience, 1981.

[30] University of Southern California Information Sciences Institute. The network simulator - ns-2.
`http://www.isi.edu/nsnam/ns/`.

[31] M. Irani, P. Anandan, and S. Hsu. Mosaic based representations of video sequences and their applications. In *Proceedings of Fifth International Conference on Computer Vision (ICCV'95)*, pages 605–611, June 1995.

[32] T. Kadir, A. Zisserman, and M. Brady. An affine invariant salient region detector. In *Proceedings of Eighth European Conference on Computer Vision(ECCV 2004)*, pages 228–241, May 2004.

[33] S. B. Kang, R. Szeliski, and M. Uyttendaele. Radial distortion snakes. *IEICE Transactions on Information and Systems*, E84-D(12):1603–1611, December 2001.

[34] Syed Ali Khayam. The discrete cosine transform (dct): Theory and application. Technical Report ECE 802-602, Michigan State University, 2003.

[35] A. Levin, A. Zomet, and Y. Weiss. Seamless image stitching in the gradient domain. In *Proceedings of IEEE Society on Computer Vision and Pattern Recognition*, pages 306–313, June 2004.

[36] T. Lindeberg. Scale-space for discrete signals. *IEEE transactions on Pattern Analysis and Machine Intelligence*, 12(3):234–254, March 1990.

[37] C. Loop and Z. Zhang. Computing rectifying homographies for stereo vision. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition(CVPR'97)*, pages 125–131, June 1999.

[38] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60:91–110, November 2004.

[39] B. D. Lucas and T. Kanade. An iterative image registration technique with an application in stereo vision. In *Proceedings of the 1981 DARPA Image Understanding Workshop*, pages 121–130, April 1981.

[40] M.El Melegy and A. Farag. Nonmetric lens distortion calibration: Closed-form solutions, robust estimation and model selection. In *Proceedings of Ninth International Conference on Computer Vision(ICCV'03)*, pages 554–559, October 2003.

[41] K. Mikolajczyk and C. Schmid. Scale and affine invariant interest point detectors. *International Journal of Computer Vision*, 60(1):63–86, October 2004.

[42] H. Moravec. The stanford cart and the cmu rover. In *Proceedings of the IEEE*, pages 872–884, July 1983.

[43] S. Nene and S. K. Nayar. A simple algorithm for nearest neighbor search in high dimensions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(9):989–1003, September 1997.

[44] D. Nister and H. Stewenius. Scalable recognition with a vocabulary tree. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 2161–2168, June 2006.

[45] International Organization of Standardization. Mpeg-2. `http://www.chiariglione.org/mpeg/standards/mpeg-2/mpeg-2.htm`.

[46] International Organization of Standardization. Mpeg-4. `http://www.chiariglione.org/mpeg/standards/mpeg-4/mpeg-4.htm`.

[47] M. Okutomi and T. Kanade. A multiple baseline stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(4):353–363, April 1993.

[48] P. Perez, M. Gangnet, and A. Blake. Poisson image editing. *ACM Transactions on Graphics*, 22(3):313–318, July 2003.

[49] K. R. Rao and P. Yip. *Discrete consine transform: algorithms, advantages, applications.* Academic Press Professional, Inc., 1990.

[50] Peter J. Rousseeuw and Annick M. Leroy. *Robust Regression and Outlier Detection.* Wiley-Interscience, 2003.

[51] D. Santa-Cruz, T. Ebrahimi, J. Askelof, M. Larsson, and C. A. Christopoulos. Jpeg 2000 still image coding versus other standards. Technical Report ISO/IEC JTC1/SC29/WG1 N1816, Ericsson Research, Corporate Unit, S-164 Stockholm, Sweden, July 2000.

[52] H. S. Sawhney and R. Kumar. True multi-image alignment and its application to mosaicing and lens distortion correction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(3):235–243, March 1999.

[53] F. Schaffalitzky and A. Zisserman. Multi-view matching for unordered image sets, or how do i organize my holiday snaps? In *Proceedings of Seventh European Conference on Computer Vision*, pages 414–431, May 2002.

[54] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision*, 47(3):7–42, May 2002.

[55] J. Shi and C. Tomasi. Good features to track. In *Proceedings of Ninth International Conference on Computer Vision and Pattern Recognition(CVPR'94)*, pages 593–600, June 1994.

[56] G. Stein. Lens distortion calibration using point correspondence. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition(CVPR'97)*, pages 602–608, June 1997.

[57] Charles.V. Stewart. Robust parameter estimation in computer vision. *SIAM Review*, 41:513–537, September 1999.

[58] R. Szeliski. Video mosaics for virtual environments. In *Proceedings of IEEE Computer Graphics and Applications*, pages 22–30, March 1996.

[59] Richard Szeliski. *Image Alignment and Stitching: A Tutorial.* Now Publishers, 2006.

[60] D. S. Taubman and M. W. Marcellin. *JPEG2000 - Image Compression Fundamentals, Standards and Practice.* Springer, Kluwer, Dordrecht, 2002.

[61] Wikipedia the free encyclopedia. Jpeg. `http://en.wikipedia.org/wiki/JPEG`.

[62] Q. Tian and M. N. Huhns. Algorithms for subpixel registration. *Computer Vision, Graphics, and Image Processing*, 35(2):220–233, August 1986.

[63] B. Triggs. Detecting keypoints with stable position, orientation, and scale under illumination changes. In *Proceedings of Eighth European Conference on Computer Vision(ECCV 2004)*, pages 100–113, May 2004.

[64] M. Uyttendaele, A. Eden, and R. Szeliski. Eliminating ghosting and exposure artifacts in image mosaics. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 509–516, December 2001.

[65] Gregory K. Wallace. The jpeg still picture compression standard. *IEEE Transactions on Consumer Electronics*, 38(1):28–34, February 1992.

[66] Y. Weiss. Deriving intrinsic images from image sequences. In *Proceedings of Eighth International Conference on Computer Vision*, pages 7–14, July 2001.

[67] I. Zoghlami, O. Faugeras, and R. Deriche. Using geometric corners to build a 2d mosaic from a set of images. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition(CVPR'97)*, pages 420–425, May 1997.