ABSTRACT

Title of Document:	VIDEO TEMPLATE MATCHING ALGORITHM FOR CONSTRUCTION PROJECTS---A HADAMARD DOMAIN APPROACH
	Xue YANG, Master of Science 2014
Directed By:	Professor Ali Haghani Department of Environment and Civil Engineering

One exciting prospect of modern construction projects is the potential for multimedia techniques, such as real-time video, to significantly affect the way of project delivery. Of all techniques in real-time video processing, template-matching plays the most essential role because of its high computational complexity and its ability to deal with considerable redundancy.

However, commonly used template-matching techniques in spatial domain cannot meet all of the requirements for all construction applications. Some methods have heavy computational burden, others suffer from inadequate accuracy. Therefore, an adjustable template-matching capable of meeting all requirements is an exciting prospect.

The proposed template-matching algorithm utilizes special relation between associated Hadamard determinants. Results indicate the proposed algorithm outperforms many popular algorithms without increasing computational complexity

level. Moreover, the algorithm is capable of adjusting three parameters accordingly to meet different construction-related applications.

VIDEO TEMPLATE MATCHING ALGORITHM FOR CONSTRUCTION PROJECTS---A HADAMARD DOMAIN APPROACH

By

Xue YANG

Thesis submitted to the Faculty of the Graduate School of the
University of Maryland, College Park, in partial fulfillment
of the requirements for the degree of
Master of Science
2014

Advisory Committee:
Professor Ali Haghani, Chair
Professor Qingbin Cui
Professor Minqi Wang

Acknowledgements

This project is a combination of the effort of many people. I owe the deepest gratitude to my supervisor Professor Ali Haghani, for his support, instruction, guidance and help. His invaluable instruction and guidance helped me to think about my work in new and different ways. I also would like to thank my thesis committee Professor Qingbin Cui and Professor Minqi Wang for their great patience and support for the work.

Table of Contents

List of Tables

List of Figures

Chapter 1: Introduction

Background

The construction industry is one of the largest employers in United States. Construction projects often face certain problems, some of which include a high accident rate, low productivity, stolen materials, stakeholders complaining about insufficient information, legal disputes, miscommunication between home office and on-site project managers, and unreasonable mobilization. Presumably, these problems arise because construction projects often involve a number of different parties working in parallel to undertake and complete a number of different tasks. Often, no crystal-clear, coherent and efficient (real-time) information is exchanged.

To effectively and efficiently deliver construction projects and overcome these common problems, one commonly used tactic is to make use of information technology by recording video on jobsites.

As the development of modern video applications and wireless communication techniques progress, more and more IP cameras are installed on jobsites to monitor human safety, laborer behavior and other purposes during construction. Also, as construction projects become more complex, the impossibility of traditional monitoring methods, which rely on a project manager's comprehensive knowledge of project details and his or her ability to control individual task progress, has become apparent. Alternatively, wireless video can be used to help project managers better control their projects. For instance, owner can see what is happening on the jobsite 24/7 even if he is far away; investors may watch their money grow and get access to any of their sites by cell phone in the palm of their hands; project manager will be able to monitor the jobsite at any given time---like overnight while he is sleeping and those data can be served as better documentation.

Video Application for Construction Projects

Video applications used for monitoring job sites are developed and modernized daily. Advanced project management strategies have inspired multimedia-based methods of monitoring construction sites. An increasing number of construction job sites have been equipped with web cameras [1].

Web cameras are commonly installed at different locations on site to shoot video, which are transmitted through wireless bandwidth to a PC located in an on-site office. Then the video signal will be transmitted from a PC to mobile and desktop computers worldwide through a wired cable or fiber.

Listed below are some of the important video applications for construction purposes.

1) Monitoring Laborer Behavior

The project manager uses real-time video, shot with a web camera, to monitor the laborer's daily work. This video will serve as an on-site superintendent and will enhance the worker's efficiency when the project manager is absent, because the laborers' awareness that the project manager can monitor through the cameras will make productivity high and keep it high.

The on-site project manager and superintendent take advantage of watching real-time video in a job-site office. Without going out of the office, project managers can still complete their jobs with the real-time monitoring video.

"I hope the company could buy more because it really boosts the efficiency of my work," one project manager said.

Some interviews with project managers suggest they adjust quickly to the new way of monitoring a job site; many prefer to combine this technique with the traditional method.

2) Human Safety Precautions and Hazard Avoidance

Construction sites are inherently dangerous workplaces, but many accidents are foreseeable and preventable; that is, laborers may avoid injury and fatality if an experienced project manager is able to identify the risk promptly[2]. However, a project manager position is notorious for the amount of work it entails: paperwork, endless inspection jobs, and frequent communication with colleagues, stakeholders, sub-contractors and suppliers. Thus, many project managers are often unable to constantly focus on the safety of workers and other people on site. Real-time video can provide an easier and more efficient way for project managers to monitor worker safety while in their offices.

Select statistics regarding work-related fatalities, injuries, and illnesses from the United States Bureau of Labor Statistics are displayed in Table 1.1.

Table 1. 1 Work-related Fatalities, Injuries, and Illnesses in 2011[3] (thousands)

Goods producing industry	2011 Annual average employment	Total recordable cases	Percentage of total cases	Injury rate per employment
Natural resources and mining	1,644.6	65.4	8.0%	4.0%
Construction	5,576.7	250.5	30.6%	4.5%
Manufacturing	11,627.7	501.7	61.4%	4.3%
Total	18,849.0	817.6	100%	4.3%

This data reflects that more than 30 percent of injury cases in 2011 occurred in the construction industry. According to the data from the United States Bureau of Labor Statistics shown above, the fatal injury rate for the construction industry was higher in 2011 than the national average rate in this category for all industries. More specifically, the injury rate for construction laborers, which was 4.5 percent, was greater than the average injury rate for per employment.

There are three major potential dangers for workers in construction-related work, which are:

- Falls (from heights)
- Trench collapse
- Scaffold collapse

Studies show that falls from heights consistently account for the highest percentage of fatalities in the construction industry. Figure 1.1 depicts unequipped laborers working at dangerous heights. While many factors must be considered when considering falls, more than 60 percent of those factors are failures to equip a construction worker with fall prevention outfits and tools[4]. Most of these accidents end in fatalities before even being discovered by a project manager. If a project manager was able to see the undergoing work through the video and identify the potential hazard, he or she can call to halt the dangerous work immediately, preventing injury.

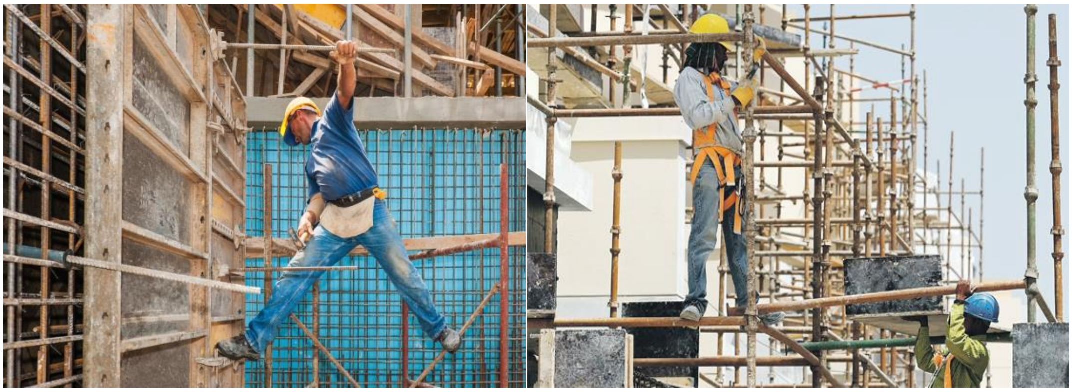


Figure 1. 1 Unequipped Construction Laborers

If project managers any dangerous practices on construction site through video, he or she should tell workers to stop working immediately. In the case that anyone, such as a stakeholder, witnesses dangerous practices using video, he or she can contact on-site superintendent or project manager to stop the unsafe practices.

In most cases, fatalities and injuries from accidents occur suddenly and unexpectedly. Therefore, it is imperative for one to stop the unsafe activity immediately to prevent accidents. Thus, monitor video must be transmitted in a real-time manner, and time delay must be avoided.

Ung Kyun Lee and Joo Heon Kim developed a mobile safety monitoring system for consruction sites[5]. Their algorithm is established on a assumption of real-time construction-site video transmission, without which the safety monitoring system cannot work effectively.

3) Theft-Proof Surveillance

The video system for monitoring worker safety may also serve as a theft-proof monitor system. Construction job sites in some areas have become increasingly subject to valuable material theft (like copper) and equipment theft (such as electric cable or even web cameras). Unfortunately, project managers and security personnel are unable to stay on site every minute. Video surveillance provides a 24-hour security monitoring of areas of interest. Moreover, certain techniques will provide real-time alerts, which can be sent to emails, cellphones, or a monitoring center for immediate assessment and response. Most surveillance services are cost-effective. To report or alert a potential theft in time, the video transmission must be in real time.

4) Remote Access to Construction Video

An added benefit to a construction site video system is that stakeholders will be able to access the video whenever they want as long as they have an Internet connection. Stakeholders can watch the project progress directly. This may prevent project managers from spending copious time and effort to present the project progress to stakeholders. As for owners, a contractor with an IP camera with remote access of on-site video will be a better candidate at private project bidding. In other words, contractor equipped with this ability has a competitive advantage.

5) Efficient Communication

Construction projects have become global. Today, more construction projects involve different teams located at many places worldwide. Real-time video transmission from one job site to another will enable talents and teams worldwide to communicate and exchange ideas freely and easily.

For education purposes, for example, it is unsafe and inconvenient for students studying construction to take frequent fieldtrips to construction sites, even though fieldtrips are beneficial. Thus, real-time video-based classes can be extremely helpful due to its simplicity and safety.

In case of problems at job sites, real-time video capabilities will enable fast and clear communication between onsite project managers and managers at home offices. Hard-to-explain concepts can be explained using video communication. Because project managers are given limited responsibilities that do not include making changes in budget, schedule, legal rights, or quality standards, for instance, they can use real-time video to communicate with a home office for instructions or delegation.. In such cases, real-time video will help project managers clarify situations and accelerate response times.

6) Project Documentation as Legal Evidence

The owner-side project manager, also called the Resident Project Representative, is responsible for maintaining a daily record of daily events, progress of construction work, laboratory test reports of materials, any changes and conversations. These records are important for legal disputes and claims. Those legal problems often arise as construction work progresses; however, it could arise even 50 years after a project's completion. A survey showed that onsite contractor personnel spend an average of 50 percent of their time recording field data, which serves as evidence in case of disputes.

With the widespread use of computers, resident project representatives can use computer memory to maintain a daily record. Because written documents prove inadequate to record all types of construction information, more emphasis has been placed upon the use of video to more accurately record construction work. Video footage can also serve as evidence to defend against potential claims. Moreover, video can be more valuable than other types of record-keeping because videos are perceived as truthful, making them stronger evidence than written records.

Video compression techniques compress high-quality video into a small data size, requiring little computer memory. However, video that serves as evidence in court requires high video quality and resolution. This requirement makes it more challenging to compress high-definition video stream into limited data.

7) Dispatch and Mobilization

Video will also prove useful in reviewing and improving construction operations, including the movement of excavators and trucks to and from the job site and dispatching the equipment more efficiently.

Mobilization includes activities like transporting prime contractors and their subcontractor's personnel, moving materials and equipment, and operating supplies to the site, establishing onsite offices and other required facilities for the operations on the site. Because construction sites have limited space, a careful deployment of personnel, materials and equipment using web camera is necessary. Otherwise, issues including safety problems may arise.

Problem Statement

Overwhelming Data Volume - Importance of Template-Matching Algorithm

The template-matching technique is a core component in video compression and video transmission due to its high computational, or temporal, complexity. The implementation of template matching has become a bottleneck in many video applications and in almost all video standards such as JPEG, MPEG, H.263 and H.264 [6]. The more effective a template-matching algorithm is, the better the overall performance could achieve.

To thoroughly understand the need for template matching, consider the amount of data of digital video in high definition (HD) standard. HD video has a higher resolution than standard definition (SD) video stream; a single HD frame has 1920×1080 pixels assuming each frame is a full-color still image. That means 24 bits is contained in each pixel. We will assume the frame rate is $30\,frame/\sec$ and video streams last one hour.

The formula calculating the size of a digital video stream is as following:

$$Raw\ video\ size = resolution \times frame\ rate \times color\ depth \times time$$

So, the raw size of digital video stream equals:

$$1920 \times 1080\,pixel/frame \times 30\,frame/\sec \times 24bit/pixel \times (60 \times 60)\sec = 5374771200000bit$$

This amount of data reaches is more than 5000 gigabytes. It is too much for both storage purposes or for transmission applications. Reducing the amount of data to a manageable level is very important [7]. For example, a flash disk with capacity one Terabyte (1024 gigabytes) is unable to store digital video without compression. However, if using a video encoder, one terabyte of memory will be able to store a 1000-hour digital video stream.

Another example is real-time video transmission. Real time video requires that a video encoder compress data and a video decoder decompress data immediately.

Since DCT, quantization and entropy coding techniques can only reduce a limited amount of data; thus, template matching is the most important when reducing redundancy within a video stream. This is because the three techniques only reduce redundancy existing in an individual frame. However, the template-matching technique is able to remove the duplication content between consecutive frames. In other words, the three techniques implement within frame dimension in the x- and y-axes, but uses template-matching process data in the temporal dimension. Figure 1.2 shows three different axes of video processing.

The demanding requirement of video storage and real-time transmission, plus limitation lying in DCT, quantization and entropy coding make template matching the most critical de-duplication techniques of all.

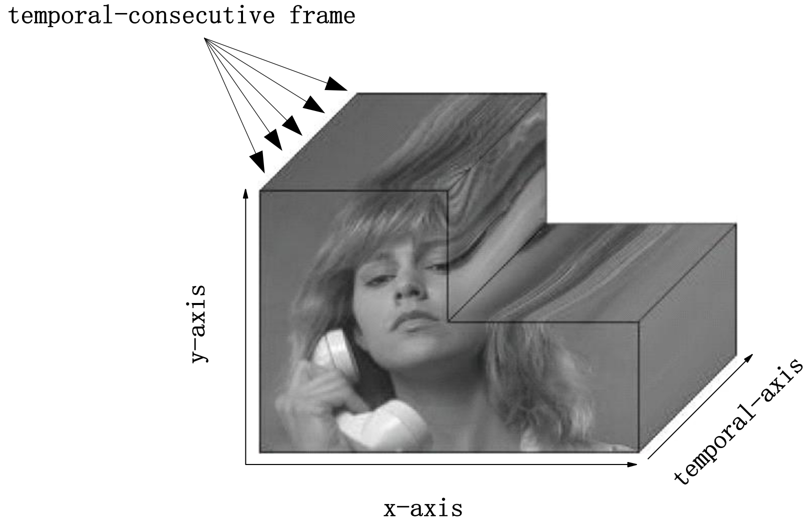


Figure 1. 2 Three Axes of Video Processing

Ineffective Template Matching --- Necessity of New Template Matching Algorithm

Template-matching techniques dominate techniques of video codec because they account for the largest computation burden. Its performance will significantly determine the performance of an entire video compression for a wide variety of applications.

But commonly used template-matching techniques cannot meet all requirements for various construction video applications. Therefore, a dynamic template-matching technique able to meet all types of diverse requirements is necessary.

Also, most of the current template-matching algorithms process video data only in spatial domain, but they are not capable of adjusting themselves to different requirements in terms of precision and timeliness. However, some algorithms in the frequency domain suffer a very high computational burden, and it may have an impact on real-time construction video transmission. So an algorithm able to process video data in frequency domain very quickly is very critical. Therefore, a dynamic and fast template matching within frequency domain is necessary.

Objectives

An overview of objectives is shown below:

As mentioned, video taken by IP camera has been used in an increasing number of applications on construction sites. The major goal of on-site video is to help deliver a smooth and safe construction workplace for both contractors and owners. Those diverse aspects include conveying clearer information in real time, monitoring job, human safety and other activities, and recording video as evidence in the event of future legal disputes.

But one of the important natures of construction video application is the wide breadth of requirements; that is, different applications of on-site video have different requirements with respect to the video properties. Unfortunately, not a single video-processing application can meet all types of requirements. Therefore a video-processing scheme able to meet or exceed all requirements of applications and scenarios is strongly needed. Moreover, video-processing frameworks (a typical video encoder) displayed in Section 2.3 is such a mature technology that a majority of 'boxes' shown in the video encoder cannot be altered or replaced, except for a template matching box. Therefore, the proposed work emphasizes developing a new template-matching algorithm. The proposed algorithm is able to meet requirements of various applications by simply adjusting three parameters.

Although some requirements are commonly shared by applications, most applications require video transmission in real time, and any time-delay will impair user experience.

Although video quality needs to be sufficient for viewing there are many other requirements for different applications. For instance, video recorded as legal evidence need not be transmitted in real time. Instead, it is most important to compress video into the least bits without decreasing the quality of any components of video. This is because original data volume, involving a great amount of detailed information, may serve as potential legal evidence and must meet those requirements; that is, poor-quality video will reduce its reliability in court.

Applications also have different precision levels, a measurement showing which level precise information is needed. If the required precise level is classified as ‘micro’, that indicates that the application demands substantive detailed and precise information. Conversely, if the precision level is classified ‘macro’, the application will only need a big picture, or the application will have a large error tolerance. For applications like remote access, precision level is classified as ‘macro’ because, in most cases, stakeholders merely want to have an overall understanding of what the progress is. When referring to theft-proof surveillance, it is important to view objects in accuracy to detect criminal activities.

Table 1.2 shows a summary of different priorities of requirements for common applications. Primary requirement is ranked highest priority since it is critical to convey information that users expect to see. Secondary requirement is less important than primary requirement, but it also helps users interpret information. Usually, most users would like to have more than one application on a construction site since they have various expectations for video about types of information and accuracy levels they want to see in different locations or scenarios. Thus, more than one video processing strategy is needed. However, those different strategies focusing on different video requirements may become cumbersome and ineffective. Also note that the most practical way to improve a video encoder framework is to upgrade template-matching boxes. So a new template matching technique, able to adapt to various scenarios in a flexible manner, is necessary.

Table 1. 2 Different Focus on Requirement of Construction Applications

Application	**Primary Requirement**	**Secondary Requirement**	**Precision Level Requirement**
Laborer Behavior Monitoring	Timely	Precise	Micro and macro
Human Safety Precaution and Hazard Avoidance	Timely	Precise	Micro and macro
Theft-Proof Surveillance	Timely	Precise	Micro
Remote Access	Precise	Timely	Macro
Efficient	Precise and timely		Macro

Communication			
Legal Evidence	Precise	Least bits (Higher compression level)	Micro
Dispatch and Mobilization	Timely	Precise	Macro

Since a frequency-domain algorithm's computational complexity is quite high, the high computational burden makes the program consume a lot of power. If IP-cameras installed at construction jobsites are in battery-constrained scenarios, emphasis must be put on reducing its power consumption. Also, high calculation burdens result in long processing times. So, a fast template-matching technique with less computational complexity is necessary.

Dissertation Organization

This dissertation is divided into seven chapters.

The first chapter provides a general introduction to the construction project's background. It enumerates seven real applications using wireless video for construction projects, and it describes some issues common to most construction projects. It also briefs the motivation of developing a novel Hadamard-domain template-matching algorithm.

Chapter 2 lays a foundation of basic concepts for video and image-processing techniques.

Chapter 3 provides a review of the most commonly-used template-matching algorithms. The review consists of two parts: exhaustive search and other fast search algorithms.

Chapter 4 describes the fundamental knowledge of Hadamard transform schemes, and it also provides information on the energy-compaction ability of Hadamard determinants. It provides a summary in terms of advancement of Hadamard transformations.

Chapter 5 explores a special relation between Hadamard determinants, which can be applied to achieve a fast template-matching algorithm. The new algorithm requires only two operations per Hadamard determinant per template matching attempt.

Chapter 6 presents the core ideas of a new template-matching algorithm, including fast calculation of cross-correlation and equivalence between cross-correlation and the Hadamard coefficient. Additionally, a fast matching criteria is developed. An implementation procedure is provided as well.

Chapter 7 presents simulation results indicating the advantages of new template-matching algorithm. Results show the new algorithm outperforms three-step search, four-step search and hexagon search practices, and it is able to closely approach the results of an exhaustive search's performance. The dissertation concludes with a summary of conclusions.

Chapter 2: Elementary Concept in Video Processing

There are several principle applications of video processing: video storage, video transmission and machine perception. Because ‘a picture is worth a thousand words,’ more construction projects are beginning to use video assistance for construction jobs. The dissertation focuses on video transmission and its application on construction site.

Digital video consists of a set of digital images, which are also known as frames. The relation between a video stream and its frames is displayed in Figure 2.1. Each digital image can be represented through two ways: spatial domain and frequency domain. The two methods will be discussed in the following sections.

To process image or video frame more conveniently and simply, people partition images into smaller units, which are called image block, window, template or pattern. An example is shown in Figure 2.2, in which current frame k is partitioned in several templates.

Figure 2. 1 A Series of Frames Comprising Video

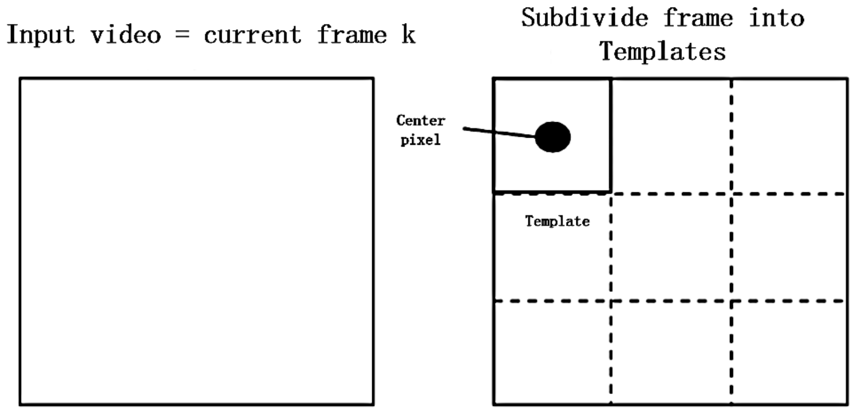


Figure 2. 2 Image and Its Templates

Digital Images in Spatial Domain

Spatial Domain Representation

An image will be first sampled and quantized in both vertical and horizontal directions to be converted into a digital image. A digital image can be denoted by a two-dimension matrix $f(x, y)$, where x and y are discrete variables, and (x, y) are discrete coordinates. For simplicity, the following will use integer values to denote (x, y): $x = 0,1,2,\ldots,M-1$ and $y = 0,1,2,\ldots,N-1$, where each frame of video is assumed to be the size of $M \times N$. Each coordinate is called a pixel or an element of an image. So, each image is formed by $M \times N$ pixels. For example, if an image is of size 500×500, there are $250,000$ pixels or elements comprising the image.

An example of representing image/frame of video in size of $M \times N$ though matrix is shown below:

$$f(x,y) = \begin{bmatrix} f(0,0) & f(0,1) & \cdots & f(0,N-1) \\ f(1,0) & f(1,1) & \cdots & f(1,N-1) \\ \vdots & \vdots & & \vdots \\ f(M-1,0) & f(M-1,1) & \cdots & f(M-1,N-1) \end{bmatrix} \qquad (2.2\text{-}1)$$

Each element of $f(x,y)$ carries information for each dot of the image by representing the brightness and chromatic information.

Spatial and Intensity Resolution

Spatial resolution is a measure of the smallest discernible units of an image. One of the most widely used measures of spatial resolution is dots per inch, or dpi, which is the number of pixels per inch of an image. Higher spatial resolution will provide a better expression of an image. However, stating only a pixel number without referring its corresponding spatial length will not provide any useful information of the spatial resolution of given image.

Intensity resolution represents the amount of intensity levels of an image. For the reason of hardware, its value usually is an integer power of two[8]. Similarly, higher intensity resolution will result in better description of given image.

As an example, a monochrome image may be represented as 1024×1024dpi, and 8bits intensity resolution per pixel. A three-color based image may be represented as 1024×1024dpi, but 8 bits per color per pixel. Since it has a three-color basis, the image's overall intensity resolution is 24 bits per pixel.

Digital Images in Frequency Domain

Image Transform Schemes

In some applications, images are converted to frequency domain prior to being processed in frequency domain, instead of being processed directly in spatial domain, There are several widely used transformation techniques, such as Discrete Fourier Transform (DFT), Discrete Cosine Transform (DCT), and Hadamard Transform (WHT).

These transformation techniques are very important to engineering applications. For example, DCT converts a finite sequence of discrete data points to a set of real sinusoidal functions centered at different frequencies. It is usually used in 'lossy' compression of audio and images as a pre-process of original data. Applications such as MP3, WMV, JPEG and MPEG standards utilize DCT to re-assign energy distribution of data points[9] and only keep low-frequency components that account for most energy, discarding the small high-frequency components. As for DFT, the most well-known discrete transform, it expresses a finite sampled function in terms of a set of coefficients and its corresponding complex sinusoids, arranged by oscillating frequencies, in an increasing order[10]. DFT converts original spatial domain data into a frequency domain, and the coefficients of different frequencies represent transformed data in the frequency domain.

Hadamard transform is an orthogonal and symmetric transform[11]. It converts a sequence of spatial domain data points to frequency domain. Unlike DCT and DFT, WHT does not have a number of clear-defined oscillating frequencies; it instead uses 'sequency' to substitute 'frequency.' It is usually utilized in video compression technique, such as MPEG-4 and H.264 standard[12] [13].

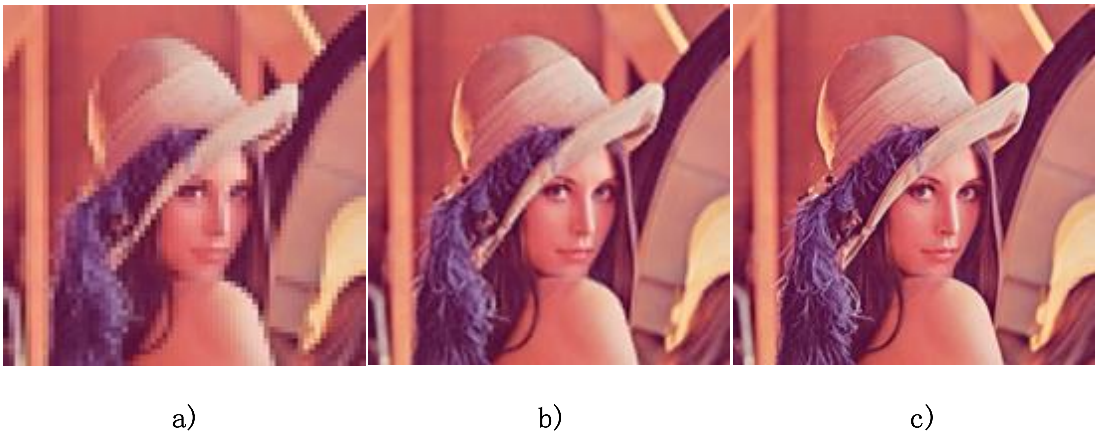
Hadamard transform has been used by NASA as a basis for compressing photographs from its interplanetary probes. Because a Fourier transform requires substantive multiple and divide operations, which are extremely time-intensive on the small computers used on spacecraft, NASA requires a substitute to avoid Fourier transform. Hadamard transform is a computational substitute for the Fourier transform, because Hadamard transform requires no multiplication or division operations; instead, all operations are subtraction or addition. This is effective because the Hadamard transform has a rectangular-shaped transform function and its factors are either plus one or minus one. Fourier transform, in contrast, has sinusoidal function, which means it is more time-consuming when computing coefficients in the frequency domain. Thus, using a Hadamard transform is beneficial both in terms of computing time and energy consumption. In the following chapter, the Hadamard transform will be discussed in detail. .

An Example of DCT

As mentioned previously, DCT has been applied widely in video compression. The original video data is represented by a sequence of data points in the spatial domain. The Discrete Cosine Transform converts data points from the spatial domain to the frequency domain. Thereafter, the transformed frequency-domain video data will be transmitted from antenna at transmitter side to antenna at receiver side through a wireless channel.

The original video stream will be converted to frequency domain by DCT, and then frequency-domain data will be transmitted from antenna to the receiver. At the receiver, data will be processed by IDCT and will return to spatial domain video stream as output.

The following figure shows how DCT will affect the quality of video sequence. Image a) is reconstructed by only 1% of total DCT coefficients. image b) is reconstructed by 3% coefficients only; and c) uses 35% of coefficients as resources for reconstruction. It is clear image b) is sufficiently clear for human eyes to perceive. It contains sufficient detail level for human eyes to interpret. Moreover, images c) and b) do not have a lot of very distinct pixels, except for those representing trivial detail.



a) b) c)

Figure 2. 3 Reconstructed Images Based on Different Amount of DCT Coefficients

Typical Video Encoder

To have a better comprehension for a transformation scheme, video transmission and its compression technique, an example is provided.

A typical video compression encoder is shown in Figure 2.4[14]. This encoder follows H.264 standard. The Transform block and Inverse Transform block represent DCT and inverse DCT (IDCT) operations. In the Patten Matching block, a Hadamard transform may be used to get a better performance.

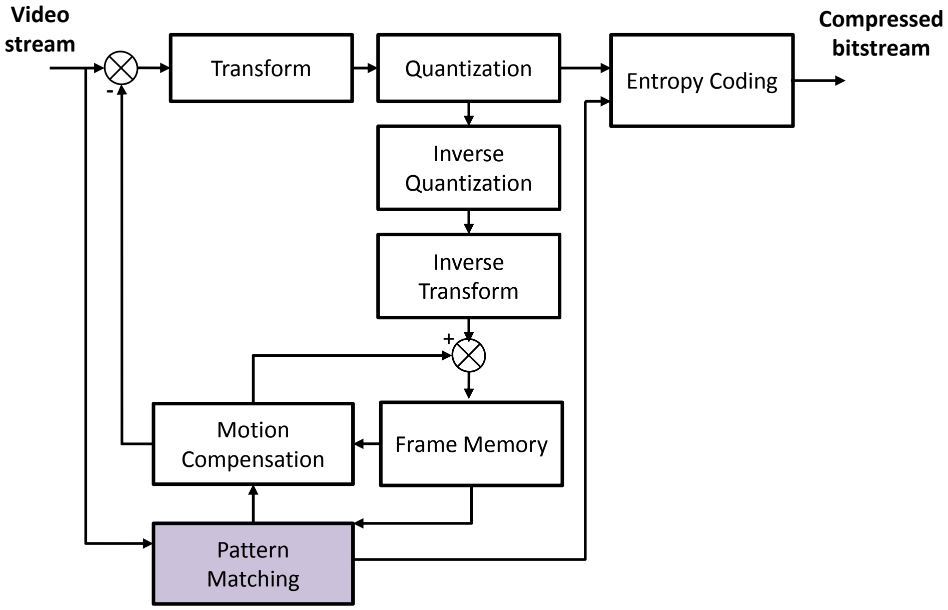


Figure 2. 4 Video Compression Structure

A video encoding process consists of several steps:

— **Subdivision**

The original video frame usually is large: for example, 1024×1024. It is difficult to process such amount of data at the same time. The most common way is to first subdivide the current frame into a number of squared sub-images in similar sizes of 16×16. (As mentioned previously, those sub-images are also called templates, windows, blocks or macroblocks.)

Therefore, each block consists of 16×16 pixels only. It is easier to process video stream on a block-by-block basis. A block is the unit of video processing and is the input of video encoder.

— **Discrete Cosine Transform**

Each data point within a block is converted to frequency-domain representation, using two-dimensional Discrete Cosine Transform. Each representation consists of a coefficient and its corresponding cosine function oscillating at a unique frequency. The larger values converge to the upper-left corner of frequency-domain point, concentrating most of the energy of the original spatial-domain block. Discrete Cosine Transform reassigns the energy distribution of the image block. Energy distribution is changed from uniform distribution to a new distribution where the top left corner captures the most energy. Because entries at the top-left corner of the block are associated with lower frequencies, Discrete Cosine Transform actually concentrates most video energy to very few data points located at top-left corner of frequency-domain block.

Two-dimensional Discrete Cosine Transform can be represented by:

$$C_{x,y}\left[u,v\right] = \begin{cases} \sum_{n=0}^{N-1}\sum_{m=0}^{N-1} x\left[n,m\right]\cos\left(\frac{\pi}{N}u(n+\frac{1}{2})\right)\cos\left(\frac{\pi}{N}v(m+\frac{1}{2})\right), & 0 \le u,v < N \\ 0 & otherwise \end{cases} \tag{2.2-2}$$

where $x\left[n,m\right]$ denotes each pixel of spatial-domain image, $C_{x,y}\left[u,v\right]$ is the output element in frequency domain (also called the transform coefficient) and image block is in size $N \times N$, usually $N = 16$, in some cases $N = 8$.

Figure 2.5 presents Discrete Cosine transform and Inverse transform between spatial-domain data points and converted frequency-domain coefficients. The red arrow starts from low-frequency coefficients with the most energy concentrated and points to high-frequency coefficients with less energy.

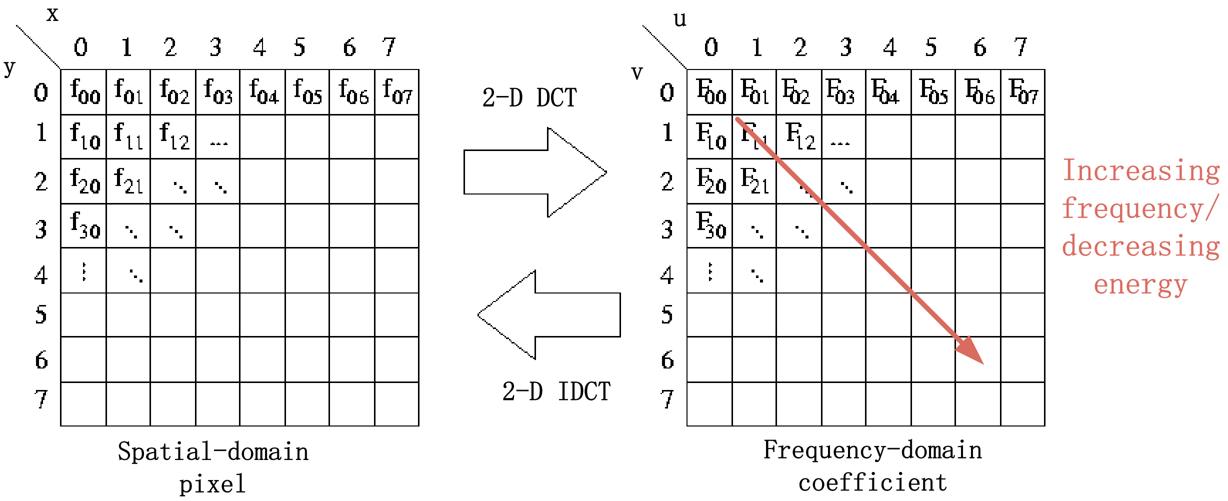

x
y
0 1 2 3 4 5 6 7
f00 f01 f02 f03 f04 f05 f06 f07
f10 f11 f12 ...
f20 f21
f30
2-D DCT
2-D IDCT
u
v
F00 F01 F02 F03 F04 F05 F06 F07
F10 F11 F12 ...
F20 F21
F30
Increasing frequency/ decreasing energy
Spatial-domain pixel
Frequency-domain coefficient

Figure 2. 5 DCT and IDCT

As an example, a 8×8 spatial-domain image block is denoted by matrix $A_{8\times8}$. For simplicity, this block consists of value 100 only.

$$A_{8\times8}=\begin{bmatrix}100 & 100 & \cdots & 100\\ 100 & \ddots & & 100\\ \vdots & & & \vdots\\ 100 & \cdots & & 100\end{bmatrix}$$

As $N=8$, a two-dimensional transform formula can be represented as

$$C_{x,y}[u,v]=\begin{cases}\sum_{n=0}^{7}\sum_{m=0}^{7}x[n,m]\cos\left(\frac{\pi}{8}u(n+\frac{1}{2})\right)\cos\left(\frac{\pi}{8}v(m+\frac{1}{2})\right), & 0\le u,v<8\\ 0 & otherwise\end{cases}$$

After calculating coefficients using the previous formula, the transformed coefficient block, denoted by $B_{8\times8}$, the frequency-domain coefficient block is:

$$B_{8\times8}=\begin{bmatrix}800 & 0 & \cdots & 0\\ 0 & 0 & & 0\\ \vdots & & \ddots & \vdots\\ 0 & 0 & \cdots & 0\end{bmatrix}$$

All coefficients except the most top-left element are converted to zero. This means most energy is converged to the lowest-frequency components.

— **Quantization**

DCT reassigns energy and converges most energy to very few coefficients with larger values at upper-left side of coefficient block. Then, quantization takes it a step further. Quantization further compresses the value of high-frequency coefficients toward zero, using a quantization matrix, and rounds the result to the nearest integer, zero.

After quantization, most bottom-right side coefficients will converge to zero. Those zero-value coefficients will be removed prior to being sent to the next processing step. Note that unlike DCT or entropy coding, quantization is irreversible, which is because coefficients close to zero have been rounded to zero. Since quantization is irreversible operation, it is a 'lossy' compression technique as well.

Because human eyes are more sensitive to low-frequency components of images and less sensitive to high-frequency components, the loss of very high frequency components of an image will not change what human eyes perceive. Quantization takes advantages of the characteristics of human eyes by simply discarding bottom-right entries of frequency-domain block representing high frequency components of an image. The result suggests that discarding high-frequency components of an image will drop the overall video quality slightly.

— **Entropy Coding**

Entropy is the smallest amount of bits needed to represent data. Thus, it is rather wasteful to use more bits than its entropy to represent data. There are several commonly used coding methods focusing on expressing full information contained in data using as few bits as possible. For example, if a pixel's entropy equals to 4 bits, 4 bits can represent this pixel because the lower bound of representing pixel is 4 bits. To effectively compress the data being transmitted, the representation of this pixel should not be more than 4 bits. Otherwise, longer time or larger bandwidth is required. Also, entropy coding is a lossless video compression technique.

— **Template Matching**

Video frame rates are high - usually over 24 frames per second – and objects or people in video do not move very quick or vey abruptly. Consecutive frames in video streams have a lot in common[15]. Two consecutive frames are displayed in Figure 2.6, as mentioned previously, and the background of two images look almost the same except for slight difference at the woman's face, as the green box indicates. But transmitting same information more than once is time-consuming and not cost-effective.

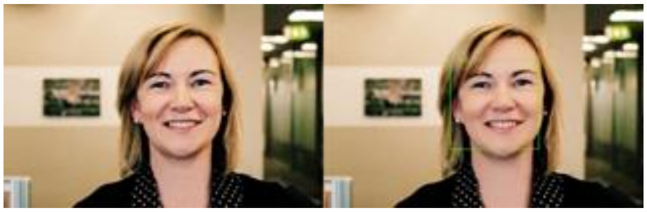


Figure 2. 6 Consecutive Frames

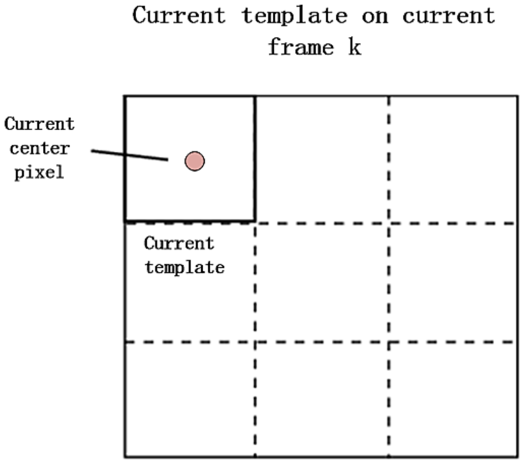
To remove redundancy among consecutive frames, one can subtract the current frame from its previous frame and transmit their differences only. It is a useful method but not as effective as expected. Template Matching and Video Compensation techniques are necessaryto further compress video in less data.

Since the video stream is being processed on a block-by-block basis, the Template Matching algorithm first has a current template located at the current frame as reference, and then compares it against a number of image blocks with the same size as the template on previous frames. The candidate image block is also called a candidate window. All windows are chosen from a pre-defined search region. Among all candidate windows, the window most similar to the template will be selected as the best-matched block. Their similarity is measured by difference between the template and the windows. Some commonly used similarity measuring metrics can be classified in two categories: spatial-domain measure, such as Sum of Absolut Differences

(SAD), and frequency-domain measure, like Sum of Absolut Transformed Differences (SATD). All measuring metrics will be discussed in the next section.

In other words, Template Matching compares and examines the likeness of consecutive frames on a block-by-block basis[16] and selects the best-matched block against the template. Consequently, Video Compensation and subtraction operations will remove redundancies between a template and its best-matched window. Therefore, the only difference between two most similar image blocks, which are the current template and its best-matched block, will be sent to DCT, quantization and entropy coding operations, and then will be transmitted to the receiver.

Figure 2.7 explains the relation between the template block and its search region, corresponding candidate windows, and the best-matched block.



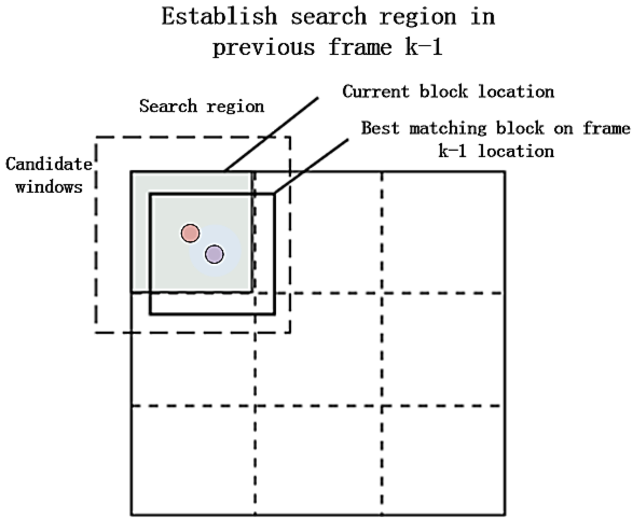

Establish search region in previous frame k-1
Search region
Current block location
Best matching block on frame k-1 location
Candidate windows

Figure 2. 7 Template, Candidate Windows and Best-matched Block

The fundamental steps of Template Matching include:

- Pre-defining a Search Region on previous frames for template blocks located on the current frame. Search regions usually surround similar locations on templates and on previous frames.
- Compare similarities between templates and all candidate windows in the Search region[17].
- Select the window with the minimum difference compared against the template as a best-matched block.

Since each comparison is based on a block-by-block basis, template matching is also named block-based template matching. The process of block-based template matching is shown in Figure 2.8.

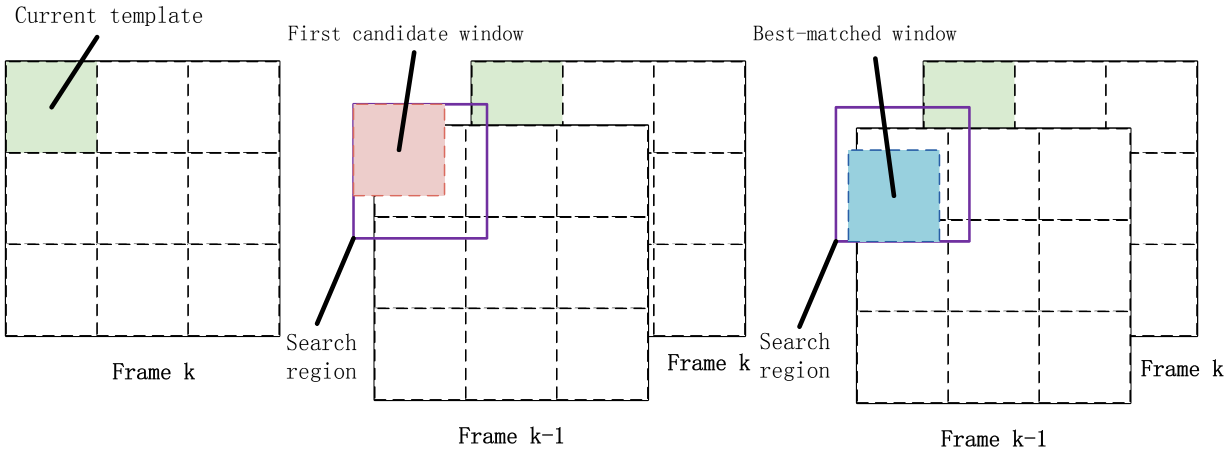


Figure 2. 8 Template Matching Process

Similarity Measures

Sum of Absolute Differences

The Sum of Absolute Differences, also known as SAD, is one common algorithm for measuring similarities between two given images[18]. It works by taking the absolute difference between each pixel in the original image block, also called the template, and its corresponding located pixel in the block used as a reference for comparison, which is called the candidate window. Each template will have a number of candidate windows for comparison. Thus, for each template, there is same number of SAD results associated with candidate windows. Among these windows, the one with the smallest result value is called the best-matched block with respect to the original template.

An instance of the calculation of SAD is shown in Figure 2.9. The template is a 3×3 block, which is denoted by a 3×3 matrix. Since search area is 4×4, there is a total of four candidate windows, and each can be obtained by sliding the blue box throughout Search region. In Figure 2.8, the first candidate window is surrounded by the first blue box.

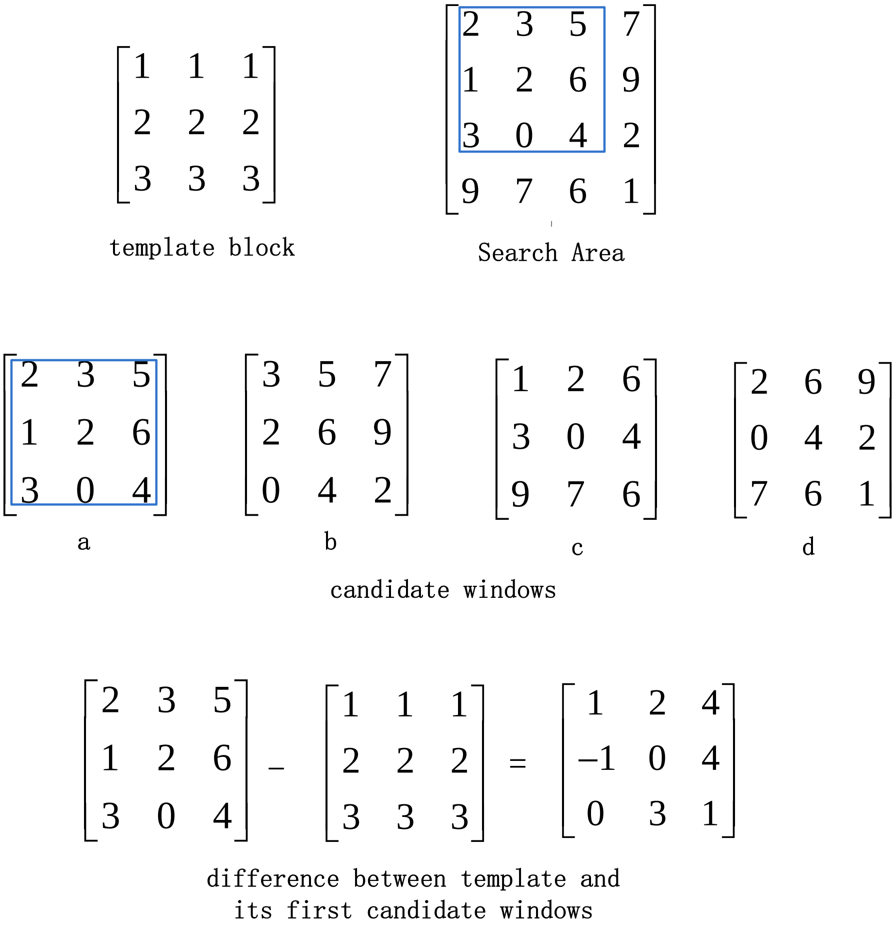


Figure 2. 9 Calculation of SAD

Therefore, the likeness index of the template and the first candidate window, measured by the Sum of Absolute Difference, can be calculated as:

$$SAD = 1 + 2 + |-1| + 0 + 4 + 0 + 3 + 1 = 12$$

Sum of Squared Differences

The Sum of Squared Differences, or SSD, is also a widely used algorithm that measures the similarity or difference between two given images. It is calculated by taking the square operation of deviation between two blocks of the two images. This measure is based on probability theory and statistics, where a random variable's variance is defined as the squared deviation of the random variable's expected value from its mean[19].

Sum of Absolute Transformed Differences

The Sum of Absolute Transformed Differences, or SATD, is a commonly used video frame similarity measuring metric[18]. It works by first projecting two image blocks into frequency domain, taking a Hadamard Transform (usually), and then calculating the difference between the transformed image blocks instead of the original spatial domain image blocks. One can calculate the difference by conducting subtraction operations between pixels in a template block and its corresponding candidate windows, and then adding the absolute values of the results together. Unlike SAD or SSD, which is similarity measuring metric in spatial domain, SATD is a measure in the frequency domain.

Summary of Measuring Metrics

There are some other algorithms measuring the similarity between two images, such as Zero-Mean Sum of Absolute Difference (ZSAD) and Zero-Mean Sum of Squared Difference (ZSSD). Table 2.1 shows the commonly used algorithms and their formulas. I denotes the frame of video, and $I(x, y)$ denotes each pixel of the image block. For convenience, the partition of frame into image blocks is not considered.

Table 2. 1 Commonly Used Algorithms

Similarity Measuring Metric	Formula	Domain
Sum of Absolute Differences	$\sum_{(i,j)\in W} \lvert I_1(i,j) - I_2(x+i, y+j) \rvert$	Spatial
Sum of Squared Differences	$\sum_{(i,j)\in W} \left(I_1(i,j) - I_2(x+i, y+j)\right)^2$	Spatial
Sum of Absolute Transformed Difference	$\sum_{(i,j)\in W} \lvert \hat{I}_1(i,j) - \hat{I}_2(x+i, y+j) \rvert$	Frequency
Zero-Mean Sum of Absolute Difference	$\sum_{(i,j)\in W} \lvert I_1(i,j) - \bar{I}_1(i,j) - I_2(x+i, y+j) + \bar{I}_2(x+i, y+j) \rvert$	Spatial
Zero-Mean Sum of Squared Difference	$\sum_{(i,j)\in W} \left(I_1(i,j) - \bar{I}_1(i,j) - I_2(x+i, y+j) + \bar{I}_2(x+i, y+j)\right.$	Spatial

Chapter 3: Literature Review

A large number of template-matching techniques have been developed to suit many construction applications. As mentioned in previous sections, most template-matching techniques are designed to process video data in the spatial domain for simplicity. But very limited amounts of techniques will project spatial-domain data to frequency domain and then process data in the frequency domain. This is because the projection (or conversion) operation will take extra time and more computational efforts; however, the performance of processing video in frequency domains is almost the same as performance in the spatial domain. There is no significant improvement in performance. Consequently the mainstream put more emphasis on directly processing received video in the spatial domain.

There are two major categories of template-matching implementations in the spatial domain, which are:

- Exhaustive search
- Fast search algorithms

Exhaustive Search

The exhaustive search algorithm is recognized by its simplicity and brutality. Its implementation exhaustively searches all possible candidate windows on the entire reference frame (usually the previous frame) and makes comparisons against the template on the current frame. Some standards also refer to this technique as the full search technique[20]. The benefit of this technique is obvious: since every possible candidate window is being considered and being used for reference when comparing against the template, the global minimal difference measuring can be successfully found. This process suggests the technique will provide accurate results. The drawback of the exhaustive search is also quite understandable: because every single window is being searched to calculate the difference between the template blocks, its computational complexity is significant, and a large run-time is rendered. In the event of limited bandwidth, the video transmission may become non-real time, or it may even suffer an abrupt long-time image stuck (which means the video is 'frozen' at the receiver side and stop at one frame for a long time).

To get a thorough understanding of the computational complexity, an analysis is given: Assume frame size is $N_1 \times N_2$, and each image block size is $N \times N$. Also, assume difference-measuring criterion is defined by SAD. Recall that $SAD = \sum_{(i,j)\in W} \left| I_1(i,j) - I_2(x+i, y+j) \right|$. To search the best-matched block for each template, every search attempt needs N^2 subtraction operations, N^2 absolute operations and $N^2 - 1$ addition operations. Therefore, in terms of each candidate window, the computational complexity is $(3N^2 - 1)$ [21]. Since all candidate windows on the reference frame size of $N_1 \times N_2$ are being considered, there are $N_1 \times N_2$ candidate windows in all. If each operation requires a unit of computational effort, the entire complexity for each template will be $N_1 \times N_2 \times (3N^2 - 1) \propto O(N^2 N_1 N_2)$ units. And for each frame, since there are $\frac{N_1 N_2}{N^2}$ template blocks in all, corresponding complexity in terms of each frame is $N_1 \times N_2 \times (3N^2 - 1) \times \frac{N_1 N_2}{N^2} \propto O(N_1{}^2 N_2{}^2)$. Exhaustive search is the most computationally expensive template-matching algorithm among all. The advantage is its accuracy-associated performance is the best of all.

Fast Search Algorithms

Because exhaustive search is not efficient, many algorithms have been developed to achieve a faster implementation by reducing computational efforts. The fast implementation can be classified as three major types.

A general review of three types of implementations is presented below.

1) **Inter-block motion field prediction algorithms** [22]

This approach, which incorporates the prediction of inter-block motion information, limits the unnecessary search points [23]. The fundamental idea of this algorithm is obtained from the observation that many image blocks in certain area on each frame tend to move toward the same direction and move within the same speed[24]. This observation is very important, especially when moving objects in video streams move at a not-very-high velocity, or when the object's acceleration is stable, or when the frame rate of video stream is very high [25]. In this case, there is a strong correlation among spatial adjacent blocks within each frame; sometimes this correlation may cross blocks at consecutive frames. Thus, the correlation can be in both the spatial and temporal domains. By assuming spatial and temporal-adjacent template blocks can share the same best-matched block, the template matching implementation is able to process video stream in a much more efficient way.

In other words, the key idea of algorithms in this category is taking advantage of the strong correlation among spatial and/or temporal adjacent blocks by assuming spatial and temporal adjacent templates share the same best-matched window, to reduce the unnecessary calculation in the search region [26].

2) **Variable size block matching algorithms**

This method has multiple-sized image blocks [27]. When a moving object or camera moves in high velocity or changes direction abruptly, using small-sized image block to process the video stream can have a better performance than using large-sized image blocks [28]. In contrast, if objects in video streams move at low speeds, using small-sized image block does nothing beneficial, simply wasting computational resources. Therefore, if video can be predicted to have many objects in unstable motion, or if the acceleration tends to increase and decrease often, variable-sized image blocks may be suited for these circumstances [29].

3) Reducing search position number algorithms

More than 80 percent of algorithms fall into this category. Also, more than 90 percent of standards and applications utilize algorithms of this category. Some algorithms are very common and have been used in various applications. For instance, the three-step search[30], the diamond search[31] and the four-step search reduce the amount of candidate windows and limit the size of the search region, so a relatively quick implementation can be accomplished with the sacrifice of accuracy. Four-step searches and other similar approaches are developed based on coarse-to-fine processes [32]. This process starts from a pre-defined initial search region with a large step size, then halving the step size to approach a fine and small search region.

An example of a four-step search (4SS) developed by Lai-Man Po and Wing-Chung Ma is presented below [33]

Step 1

The four step search starts by forming a search region with size of 5*5 consisting nine candidate windows, where eight windows surround the same position as current template on reference frame, and one window locates at same position as template. Calculate the difference measure criteria between nine candidate windows and current template. Select the window with minimal difference against template to become the new center of next step search region. If the window with minimal difference locates at center of current search region, jump to Step 4; otherwise go to Step 2.

Step 2

New search region remains the same size, 5 * 5 . But new search region is centered at the minimal-difference window. Other implementations are similar to Step 1.

Step 3

Repeat Step 2.

Step 4

Search region size is reduced down to 3*3. Search region is centered at window with minimal difference selected in Step 1 or Step 3. In both cases nine new candidate windows are compared against template block.

Note that Step 2 has two different scenarios. The first is when minimum-difference window locates at the vertex of search region in Step 1, and another five candidate windows need to be added to form the new search region in Step 2. The second is if the minimum-difference window locates at edge of search region in Step 1, and only three new candidate windows require consideration. Figure 3.1 displays the scenarios of Step 2.

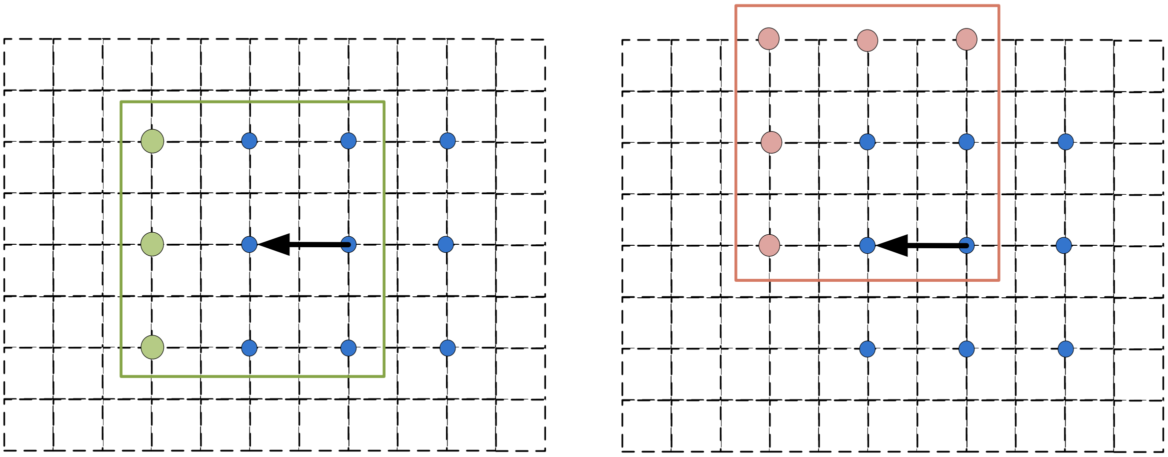


Figure 3. 1 Two Scenarios of Step 2

An explanation of a four-step search implementation is shown in Figure 3.2. Every point represents the central coordinate point of individual image block. A black point denotes the similar position to the template in the reference frame.

In Step 1, nine candidate windows, denoted by nine blue points, form the search region of 5×5. An arrow marked with "1" starts from initial template position and points to the window with minimal difference against template.

In Step 2, since previous minimum-difference window located at the edge of the previous search region, only three new candidate windows in green are being considered. Similarly, all nine windows, including three new and six old, are compared against the template block. Then, a window with the minimal difference is chosen to become new center of nest search region.

Step 3 uses another scenario. It instead considers five new windows, whose central coordinates are represented in yellow.

Step 4 has a different step-size. Its search region is 3×3. In this step, the window with the smallest difference compared to the template is selected to be the best-matched block.

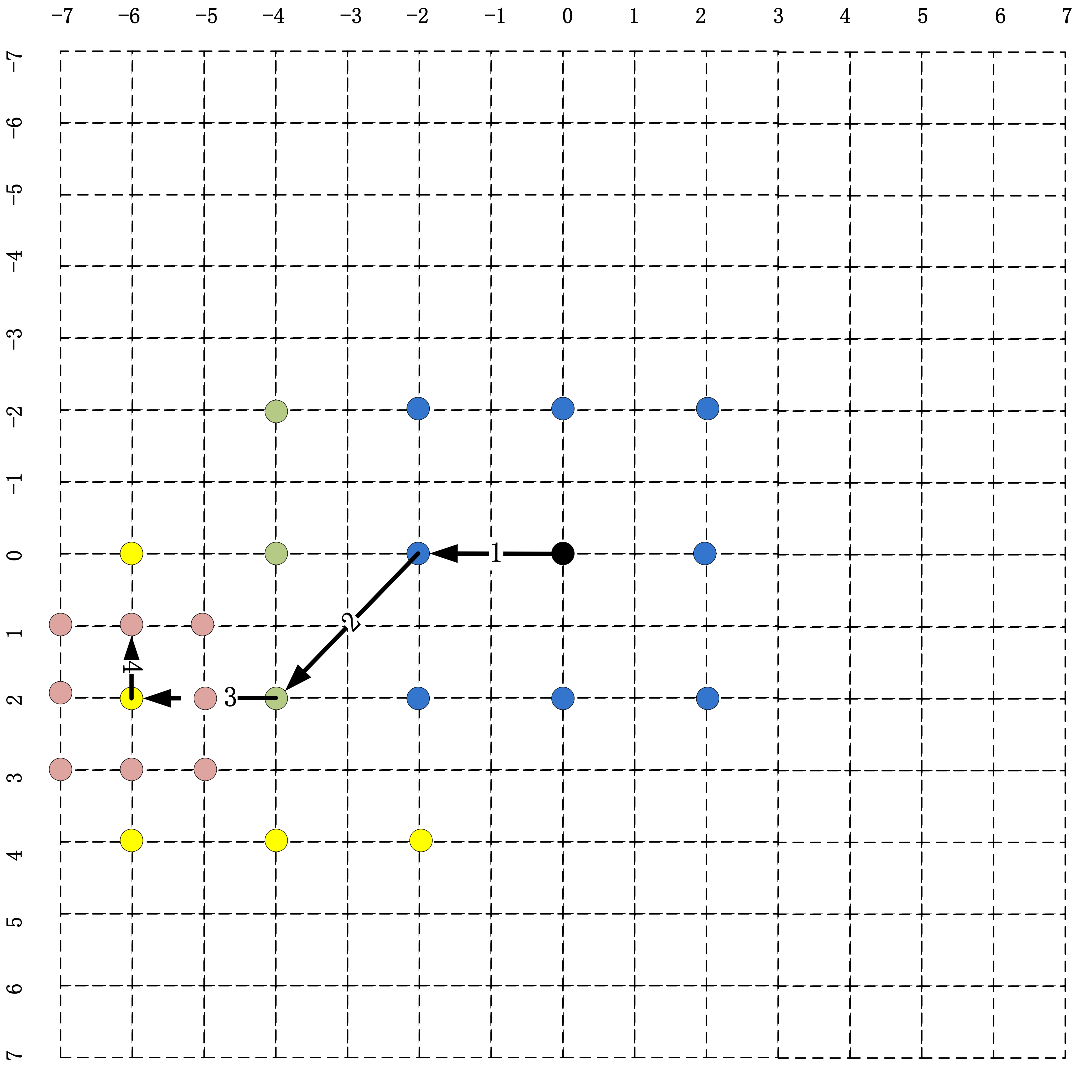


Figure 3. 2 Four Step Search Implementation

According to the implementation of the four-step search, it is obvious that the computational complexity of this algorithm varies from one case to another assuming SAD is the difference measuring criteria, each frame is $N_1 \times N_2$, and each image block size is $N \times N$.

Consider that in the simplest case that the minimum-difference window locates at the center of current search region in Step 1, it will jump directly to Step 4. Step 1 needs to calculate difference-measuring criteria nine times due to nine candidate windows being considered. Step 4 calculates the difference eight times. Thus, the total amount of candidate windows is $(9+8)=17$.

Since $SAD = \sum_{(i,j)\in W} \left| I_1(i, j) - I_2(x+i, y+j) \right|$ in terms of each candidate window, it requires $(3N^2 - 1)$ operations. So the total computational complexity for the entire frame is $17 \times (3N^2 - 1) \times \frac{N_1 N_2}{N^2} \propto O(kN_1 N_2)$, where k is a constant number. The most complex case requires $(9+5+5+8) = 27$ candidate windows per template. This is because either Step 2 or 3 has to consider another five windows in the event that the minimum-difference window is located at the vertex of the previous search region. In this case, the four-step search's complexity is $27 \times (3N^2 - 1) \times \frac{N_1 N_2}{N^2} \propto O(kN_1 N_2)$. The result is in same complex level as the simplest case. As a result, a four-step search's computational complexity is $O(kN_1 N_2)$, remarkably less than $O(N_1^{\ 2} N_2^{\ 2})$ of exhaustive search.

Chapter 4: Basic Concepts of Hadamard Transform

Basic Introduction to Hadamard Transform

As previously mentioned, the Hadamard transform has been widely utilized in video encoders and video transmission applications for video storage. A specific introduction to Hadamard transforms will be provided. The Hadamard transform will be used in the proposed template-matching algorithm to fully represent pixels in the spatial domain to the frequency (Hadamard) domain.

In the previous chapter, three major types of template-matching algorithms have been discussed. Among the three types, the type 2 algorithms are the most popular and have been applied to a wide variety of standards and applications. However, almost all template-matching algorithms process video data in the spatial domain only. Instead, there are very few algorithms developed to cope with video data in the frequency domain. Those frequency-domain algorithms are not popular compared to spatial-domain algorithms because they need to convert spatial-domain data to frequency domain at the very beginning of their implementation. Additionally, converting frames in the spatial domain to the frequency domain requires large calculation time and extensive effort, but their performance has not improved much in comparison with algorithms in the spatial domain. Due to their simplicity and good performance, spatial-domain algorithms still dominate among all template-matching techniques.

The proposed algorithm is a frequency-domain algorithm. By carefully selecting the type of transform (a Hadamard transform), we achieve a very simple and effective implementation, especially for a construction purpose. Also, by exploring the proper transform determinants selected in a pre-defined order, a unique relationship between adjacent determinants is developed. Moreover, a fast transform scheme, able to convert spatial-domain frame to frequency domain in a very fast manner, is developed by taking advantage of special relations between associated Hadamard determinants.

Hadamard Matrix, Vector and Determinant

— Hadamard matrix

Generation of a Hadamard matrix is a recursive procedure. A Hadamard matrix is a squared symmetric matrix containing only two types of multipliers: $+1$ and -1. Also, every entry of Hadamard matrix is orthogonal to another [34]. The generation of a Hadamard matrix is presented.

An initial Hadamard matrix at level 0 is defined as $M^0 = [1]$. It is a 1×1 squared matrix with only one element.

To expand to larger dimensions, an expanding framework is used: [35]

$$M^n = \begin{bmatrix} M^{n-1} & M^{n-1} \\ M^{n-1} & -M^{n-1} \end{bmatrix}, \quad n = 1, 2, 3, \cdots$$

In this framework, M^n denotes a level n Hadamard matrix containing $2^n \times 2^n$ elements.

As an example, given $M^0 = [1]$

$$M^1 = \begin{bmatrix} M^0 & M^0 \\ M^0 & -M^0 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix},$$

$$M^2 = \begin{bmatrix} M^1 & M^1 \\ M^1 & -M^1 \end{bmatrix} =$$

$$= \left[\begin{array}{cc:cc} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ \hdashline 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{array}\right]$$

$$M^3 = \begin{bmatrix} + & + & + & + & + & + & + & + \\ + & - & + & - & + & - & + & - \\ + & + & - & - & + & + & - & - \\ + & - & - & + & + & - & - & + \\ + & + & + & + & - & - & - & - \\ + & - & + & - & - & + & - & + \\ + & + & - & - & - & - & + & + \\ + & - & - & + & - & + & + & - \end{bmatrix}$$

To have a clearer view of the Hadamard matrix, use $+$ to denote 1 and $-$ to denote -1. Thus, M^3 can be represented as:

$$M^3 = \begin{bmatrix} + & + & + & + & + & + & + & + \\ + & - & + & - & + & - & + & - \\ + & + & - & - & + & + & - & - \\ + & - & - & + & + & - & - & + \\ + & + & + & + & - & - & - & - \\ + & - & + & - & - & + & - & + \\ + & + & - & - & - & - & + & + \\ + & - & - & + & - & + & + & - \end{bmatrix}$$

— **Hadamard vector**

The Hadamard vector is each entry of Hadamard matrix, and each matrix M^n has 2^n vectors, each vector containing 2^n elements. For instance, M^3 consists of 8 vectors, and each vector contains 8 elements. The value in blue represents the times of sign changing, called the zero-crossing time [36]. If a vector is $[+ \quad + \quad - \quad -]$, because signs change from $+$ to $-$ once only, its zero-crossing time is 1; if a vector is $[+ \quad - \quad - \quad +]$, the zero-crossing time equals 2.

$$M^3 = \begin{array}{cccccccc} 0 & 7 & 3 & 4 & 1 & 6 & 2 & 5 \\ + & + & + & + & + & + & + & + \\ + & - & + & - & + & - & + & - \\ + & + & - & - & + & + & - & - \\ + & - & - & + & + & - & - & + \\ + & + & + & + & - & - & - & - \\ + & - & + & - & - & + & - & + \\ + & + & - & - & - & - & + & + \\ + & - & - & + & - & + & + & - \end{array}$$

To develop a fast template-matching algorithm, we rearrange the order of vectors from vector 0 zero-crossing time to vector 7 zero-crossing times. In other words, the Hadamard matrix consists of a set of vectors in increasing order of zero-crossing time. An example of the new Hadamard matrix is shown as following. It is important to note that only new Hadamard matrices will be considered in the future.

$$M^3 = \begin{array}{cccccccc} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ + & + & + & + & + & + & + & + \\ + & + & + & + & - & - & - & - \\ + & + & - & - & - & - & + & + \\ + & + & - & - & + & + & - & - \\ + & - & - & + & + & - & - & + \\ + & - & - & + & - & + & + & - \\ + & - & + & - & - & + & - & + \\ + & - & + & - & + & - & + & - \end{array}$$

We denote the vector with zero-crossing time z as v_z, where $z = 0, 1, 2, \ldots 2^n - 1$, and n represents the level of the Hadamard matrix. The new Hadamard matrix can be represented as $M^n = [v_0 \quad v_1 \quad \cdots \quad v_{2^n - 1}]$.

— **Hadamard determinant**

The Hadamard determinant is defined as the product of two Hadamard vectors [37]. The Hadamard determinant is denoted by $K_{z_1 z_2} = v_{z_1} v_{z_2}^T$. It is a $2^n \times 2^n$ squared matrix [38].

Consider an example: v_0 and v_1 can create K_{01}. Assuming the Hadamard vector is at level 2, then $v_0 = \begin{bmatrix}1 & 1 & 1 & 1\end{bmatrix}^T, v_1 = \begin{bmatrix}1 & 1 & -1 & -1\end{bmatrix}^T$. Thus, a determinant can be represented as:

$$K_{01} = v_0 v_1^T = \begin{bmatrix}1 & 1 & 1 & 1\end{bmatrix}^T \times \left(\begin{bmatrix}1 & -1 & -1 & 1\end{bmatrix}^T\right)^T = \begin{bmatrix}1 & 1 & 1 & 1\end{bmatrix}^T \times \begin{bmatrix}1 & -1 & -1 & 1\end{bmatrix}.$$

$$K_{01} = \begin{bmatrix}1\\1\\1\\1\end{bmatrix} \times \begin{bmatrix}1 & -1 & -1 & 1\end{bmatrix} = \begin{bmatrix}1 & -1 & -1 & 1\\1 & -1 & -1 & 1\\1 & -1 & -1 & 1\\1 & -1 & -1 & 1\end{bmatrix}$$

It is obvious that if vectors are in number of $2^n \times 1$, there are totally $2^n \times 2^n$ Hadamard determinants.

Figure 4.1 shows all 16 determinants generated by 4×4 vectors. White colors denote $+1$, black denotes -1.

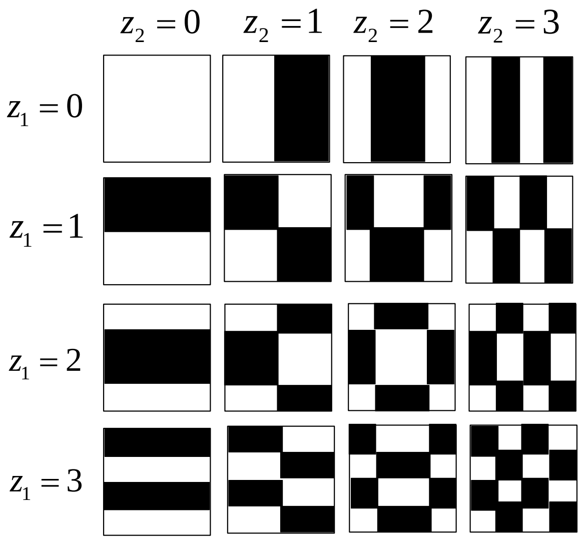


Figure 4. 1 A set of Hadamard determinants created by 4×4 Hadamard vectors

Interaction of Hadamard Matrix, Vector and Determinant

To better explore the special relation among Hadamard determinants, the relation between the Hadamard matrix, vector and determinant is studied.

As mentioned, the Hadamard matrix M^n at level n is a squared matrix containing $2^n \times 2^n$ elements in total. We denote any element of the Hadamard matrix by $M^n[x, y]$, where $x, y = 0,1,2,\ldots 2^n - 1$. Recall that each Hadamard matrix consists of 2^n vectors in size of 1×2^n, and these vectors are in increasing order in terms of their zero-crossing time. We denote the element of the vector by $v_z[x]$, where $x = 0,1,2,\ldots 2^n - 1$. As to the Hadamard determinant $K_{z_1 z_2}$, we denote elements of each determinant by $K_{z_1 z_2}[x, y]$, $z_1, z_2 = 0,1,2,\ldots 2^n - 1$ and $x, y = 0,1,2,\ldots 2^n - 1$.

Because $K_{z_1 z_2} = v_{z_1} v_{z_2}^T$, a one-to-one relation exists between elements of the Hadamard determinant and its vector:

$$K_{z_1 z_2}[x, y] = v_{z_1}[x] v_{z_2}[y] \quad (4.1\text{-}1)$$

This means any element of the Hadamard determinant can be calculated by the multiplication between an element of $v_{z_1}[x]$ and another element of $v_{z_2}[y]$.

Likewise, a relation between Hadamard matrix and vector can be denoted by:

$$M[x,y]=v_x[y]=v_y[x] \tag{4.1-2}$$

Moreover, similar relation can be found between Hadamard determinant and matrix:

$$K_{z_1z_2}[x,y]=M[x,z_1]M[y,z_2] \tag{4.1-3}$$

This relation can also be reached through the two relations shown as (4.1-1) and (4.1-2), by changing $v_x[y]$ in (4.1-2) to $v_x[z_1]$, $v_y[z_2]$.

Thus, the relation between any two parties of the Hadamard matrix, vector and determinant is shown in Table 4.1.

Table 4. 1 Relation between Hadamard matrix, vector and determinant

Parties Involved	Relation
matrix, determinant	$K_{z_1z_2}[x,y]=M[x,z_1]M[y,z_2]$
determinant, vector	$K_{z_1z_2}[x,y]=v_{z_1}[x]v_{z_2}[y]$
vector, matrix	$M[x,y]=v_x[y]=v_y[x]$

Hadamard-Domain Video Representation

Video data in the spatial domain can be converted to frequency domain using a Hadamard transform. The Hadamard transform formula is shown below:

$$W = MwM^T \qquad (4.2\text{-}1)$$

where M denotes the Hadamard matrix, w denotes the image window in the spatial domain, and $M = \begin{bmatrix} v_0 & v_1 & \cdots & v_{N-1} \end{bmatrix}$ is the transformed block in the frequency domain. M, w, W are squared matrices of sizes $N \times N$, where $N = 2^n$ and where n denotes the Hadamard matrix's level.

Element of the spatial-domain image block w are represented as $w(x, y)$, where (x, y) are the coordinates of the image block. The element of the converted image block in a frequency domain is represented as $W(u, v)$, where (u, v) are the coordinates of transformed image block. Usually, $W(u, v)$ is called the coefficient, and W is called the coefficient block. Thus, (5.1-1) can be represented by each coordinate and coefficient of the image block:

$$W[u, v] = \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} M[u, x] w[x, y] M[y, v] \qquad (4.2\text{-}2)$$

Energy Compaction Ability of Hadamard Determinant

According to filter-design techniques of digital image processing, Hadamard vector v_z with low zero-crossing times is able to compact more energy of input video data in the spatial domain. Otherwise, a vector with higher value of z compacts less energy of input data [39][40]. Likewise, this ability can be expanded to two-dimensional cases. Recall the relation between Hadamard determinant and vector, $K_{z_1 z_2}[x, y] = v_{z_1}[x] v_{z_2}[y]$, and z_1, z_2 denote the zero-crossing time of Hadamard vector. Hadamard determinant $K_{z_1 z_2}$ with smaller value of z_1, z_2 has a stronger ability to compress input data energy into Hadamard coefficients; $K_{z_1 z_2}$ with larger z_1, z_2 is unable to compress as much energy as a determinant with a smaller value of z_1, z_2.

In other words, $K_{z_1 z_2}$ with smaller z_1, z_2 is able to compact energy to fewer coefficients[41], and $K_{z_1 z_2}$ with larger z_1, z_2 needs more coefficients to represent the same amount of energy of input video. This is important when construction-related video requires various levels of precision: macro, which indicates a need for less precise information, and micro, where more precise information is needed. If the application requires macro precision level, it is wise to convert input data to few coefficients for frequency-domain template matching. Instead of converting all input data to the Hadamard domain, this finding will reduce the computational burden of converting spatial-domain data to the frequency domain.

An explanation of Hadamard representation of input video data is given as:

Since $W = MwM^T$, $M = \begin{bmatrix} v_0 & v_1 & \cdots & v_{N-1} \end{bmatrix}$, $N = 2^n$

And, since Hadamard matrix is a symmetric matrix

$$\Rightarrow M = M^T = M^{-1}$$

$$\Rightarrow w = \frac{M^T W M}{N^2} = \frac{MWM^T}{N^2}$$

$$\Rightarrow w[x,y] = \frac{1}{N^2} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} M[x,u] W[u,v] M[v,y]$$

Since the relation between the Hadamard determinant and its matrix holds:

$$K_{uv}[x,y] = M[x,u] M[y,v]$$

And since Hadamard matrix is a symmetric matrix:

$$\Rightarrow M[y,v] = M[v,y]$$

$$\Rightarrow w[x,y] = \frac{1}{N^2} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} W[u,v] K_{uv}[x,y]$$

$$\Rightarrow w = \frac{1}{N^2}\sum_{u=0}^{N-1}\sum_{v=0}^{N-1} W\left[u,v\right]K_{uv} \tag{4.3-1}$$

These equations indicate input video data can be fully represented by a series of Hadamard determinants and their corresponding Hadamard coefficients. Because Hadamard determinants have an excellent energy packing ability, a determinant with smaller u,v is able to capture more energy of input data and is able to pack the energy into its corresponding coefficient $W\left[u,v\right]$. As an example, K_{00} has the strongest energy packing ability and will store most energy of input data into $W\left[0,0\right]$.

A Hadamard transformation scheme has strong energy compaction ability. The Hadamard tansform is able to reassign energy distribution by packing the energy of input video data (spatial domain) into very few coefficients in the frequency domain. This ability is critical because it yields the possibility of discarding coefficients with little energy and focusing on most important coefficients only [42].

By applying the Hadamard transform $w = \frac{1}{N^2}\sum_{u=0}^{N-1}\sum_{v=0}^{N-1} W\left[u,v\right]K_{uv}$, it is obvious the Hadamard determinant reassigns energy from even distribution over all pixels within the frame to a new distribution that concentrate the most energy into very few coefficients. For instance, energy is originally distributed evenly over $w\left[x,y\right]$ on frame, $x,y = 0,1,2,\ldots 2^n - 1$. After a Hadamard transformation, most energy is compacted to $W\left[u,v\right]$, where u,v are very small integers.

Advantage of Hadamard Representation

The result of the Hadamard transform (HT) representation is a set of coefficients. Recall that HT will convert information in the spatial domain to the frequency domain. After applying

HT, original pixels in spatial domain are combined, re-organized and transformed into these coefficients in frequency domain.

Some advantages of HT can be summarized as following:

- Have a relation between adjacent determinants. This relation can be used in developing a very fast template-matching technique.
- Obtain an inverse operation, called the inverse Hadamard transform (IHT), so the converted data can be reverted to original data.
- Reassign each video frame's energy from evenly distributed to unevenly distributed very effectively. This will save computational effort by focusing on coefficients with the most energy and discarding others. It is more valuable for applications in which only 'macro' level of precision is needed.
- Concentrate the most energy to low-frequency representations (coefficients) of video frames, and store very little energy in its high-frequency representations. Moreover, HT's energy compaction ability proves very powerful. The more powerful the energy-compaction ability is, the fewer representations that are necessary to well-represent frame in frequency domain [43].
- Must be able to meet diverse requirements of construction applications, especially when different precision-levels are involved, because the amount of detailed and precise data can be adjusted by changing the number of high-frequency coefficients.
- Require smaller amount of coefficients to accurately describe abruptly-changed adjacent pixels within the frame. This is because the Hadamard transformation scheme has a rectangular-shaped function, but DCT and DFT have sinusoidal wave functions, and it is very difficult to represent abrupt changes through smooth functions.

- Since only real number additions and subtractions are involved in Hadamard transform, an acceleration of implementation in comparison with other common algorithms can be realized.

Chapter 5: Special Relation Exploration of Hadamard Determinants

Although frequency-domain template matching has obvious advantages over spatial-domain techniques with respect to construction applications, spatial-domain techniques are more popular than frequency-domain techniques. This is because most frequency-domain template-matching algorithms require more computational effort than spatial-domain algorithms, and it usually requires a longer processing time.

A fast frequency-domain algorithm with less computational complexity is necessary. The proposed algorithm is developed to process data in the Hadamard domain quickly on a basis of a special relation between associated Hadamard determinants. An exploration of special relation is presented in this section. The exploration starts from establishing a new generation procedure of a Hadamard matrix using a Kronecker product. Additionally, a special relation between one-dimensional Hadamard vectors is developed based on the new generation procedure. The relation is thereafter expanded to two-dimensional case between Hadamard determinants.

Kronecker Product

The definition of Kronecker product can be explained by an example.

If a $m \times n$ matrix A and $l \times r$ matrix B perform Kronecker product, the result being:

$$A \otimes B = \begin{bmatrix} \boxed{a_{11} \cdot B} & \cdots & \boxed{a_{1n} \cdot B} \\ \vdots & \ddots & \vdots \\ \boxed{a_{m1} \cdot B} & \cdots & \boxed{a_{mn} \cdot B} \end{bmatrix} \tag{5.1-1}$$

To be more specific:

$$A \otimes B = \begin{bmatrix} a_{11}b_{11} & a_{11}b_{12} & \cdots & a_{11}b_{1r} & \cdots & \cdots & a_{1n}b_{11} & a_{1n}b_{12} & \cdots & a_{1n}b_{1r} \\ a_{11}b_{21} & a_{11}b_{22} & \cdots & a_{11}b_{2r} & \cdots & \cdots & a_{1n}b_{21} & a_{1n}b_{22} & \cdots & a_{1n}b_{2r} \\ \vdots & \vdots & \ddots & \vdots & & & \vdots & \vdots & \ddots & \vdots \\ a_{11}b_{l1} & a_{11}b_{l2} & \cdots & a_{11}b_{lr} & \cdots & \cdots & a_{1n}b_{l1} & a_{1n}b_{l2} & \cdots & a_{1n}b_{lr} \\ \vdots & \vdots & & \vdots & \ddots & & \vdots & \vdots & & \vdots \\ \vdots & \vdots & & \vdots & & \ddots & \vdots & \vdots & & \vdots \\ a_{m1}b_{11} & a_{m1}b_{12} & \cdots & a_{m1}b_{1r} & \cdots & \cdots & a_{mn}b_{11} & a_{mn}b_{12} & \cdots & a_{mn}b_{1r} \\ a_{m1}b_{21} & a_{m1}b_{22} & \cdots & a_{m1}b_{2r} & \cdots & \cdots & a_{mn}b_{21} & a_{mn}b_{22} & \cdots & a_{mn}b_{2r} \\ \vdots & \vdots & \ddots & \vdots & & & \vdots & \vdots & \ddots & \vdots \\ a_{m1}b_{l1} & a_{m1}b_{l2} & \cdots & a_{m1}b_{lr} & \cdots & \cdots & a_{mn}b_{l1} & a_{mn}b_{l2} & \cdots & a_{mn}b_{lr} \end{bmatrix}$$

From this equation, it is obvious the Kronecker product is calculated by multiplying the entire matrix B to each element of matrix A [44]. Since A is $m \times n$ matrix, and B is $l \times r$ matrix, the resultant matrix $A \otimes B$ is in size of $ml \times nr$ [45].

Key Idea of Relation Exploration

The Hadamard matrix has a recursive ge neration procedure. The initial seed matrix M^0 is a 1×1 matrix $[1]$. Higher-level matrix M^n in size of $2^n \times 2^n$ is created by lower-level matrix M^{n-1} using generation formula $M^n = \begin{bmatrix} M^{n-1} & M^{n-1} \\ M^{n-1} & -M^{n-1} \end{bmatrix}$, $n = 1, 2, 3, \cdots$. Recall that the Hadamard matrix consists of a series of Hadamard vectors, $M^n = [v_0 \quad v_1 \quad \cdots \quad v_{2^n-1}]$. We denote vectors with zero-crossing time z as v_z, $z = 0, 1, 2, \ldots 2^n - 1$. The vectors are arranged in ascending order of zero-crossing time.

Illuminated by Kronecker product, a new generation procedure of Hadamard matrix is established:

$$\because M^n = \begin{bmatrix} M^{n-1} & M^{n-1} \\ M^{n-1} & -M^{n-1} \end{bmatrix} = \begin{bmatrix} 1\times M^{n-1} & 1\times M^{n-1} \\ 1\times M^{n-1} & (-1)\times M^{n-1} \end{bmatrix},\ n=1,2,3,\cdots$$

Also $\because A\otimes B = \begin{bmatrix} a_{11}\cdot B & \cdots & a_{1n}\cdot B \\ \vdots & \ddots & \vdots \\ a_{m1}\cdot B & \cdots & a_{mn}\cdot B \end{bmatrix}$

$$\therefore\ A=\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}, B=M^{n-1}$$

$\therefore$ Hadamard matrix M^n can be created by Kronecker product $A\otimes M^{n-1}$

Consider example $M^0=[1]$,

$$\Rightarrow M^1 = A\otimes M^0 = \begin{bmatrix} 1 & 1 \\ 1 & \text{-1} \end{bmatrix}$$

It is clear that by performing $A\otimes M^{n-1}$, the matrix $A=\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$ will expand the length of each Hadamard vector from 2^{n-1} to 2^n. Moreover, the resultant matrix M^n consists of two types of vectors: the first being a replication that doubles the length of the original vector, and the second being a cascade of the original vector and its opposite.

Because the Hadamard matrix is a symmetric matrix, we only need to consider its vector in vertical dimension.

Matrix $A=\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$, as mentioned, can only generate two types of Hadamard matrix, one being its replication and the other one being a cascade of the original vector and its opposite. So it is more convenient to denote it by $A=[+\quad -]$.

Consider the Hadamard matrix at level-1 M^1, since we consider vertical dimension only, M^1 can be represented as $M^1 = A \otimes M^0 = \begin{bmatrix} 1 & 1 \\ 1 & \text{-1} \end{bmatrix} \Rightarrow \begin{bmatrix} + & - \end{bmatrix}$.

Likewise, $M^2 = A \otimes M^1 = \begin{bmatrix} + & - \end{bmatrix} \otimes \begin{bmatrix} + & - \end{bmatrix} \Rightarrow \begin{bmatrix} ++ & -+ & +- & -- \end{bmatrix}$

$$
\begin{aligned}
&M^3 = A \otimes M^2 = \begin{bmatrix} + & - \end{bmatrix} \otimes \begin{bmatrix} ++ & -+ & +- & -- \end{bmatrix} \\
&\Rightarrow \begin{bmatrix} +++ & -++ & +-+ & --+ & ++- & -+- & +-- & --- \end{bmatrix} \\
&= \begin{bmatrix} v_0 & v_7 & v_3 & v_4 & v_1 & v_6 & v_2 & v_5 \end{bmatrix}
\end{aligned}
$$

After the Kronecler product, each Hadamard vector can be indicated by a series of $+$ and/or $-$. We call this series of signs ‘sign index sequence’, which can be denoted by s_z, where z is zero-crossing time of corresponding vector. For example, sign-index sequence of the Hadamard vector $v_0 = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}^T$ is $s_0 = \langle + \quad + \quad + \rangle$, sign-index of $v_7 = \begin{bmatrix} 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \end{bmatrix}^T$ is $s_7 = \langle - \quad + \quad + \rangle$.

By carefully observing the elements of any pair of vectors whose sign-index sequences only have one index inverse, a special relation between these vectors is explored.

Definition — A pair of Hadamard vectors is called ‘associated’ Hadamard vectors if their sign-index sequences only have one index different (inverse).

Between two associated vectors, we define the vector with $+$ as the only inverse index to be v^+, and we denote its sign-index sequence by s^+; otherwise, we define the vector to be v^-, whose sign-index sequence is s^-. For instance, v_0’s sign-index sequence is $s_0 = \langle + \quad + \quad + \rangle$, and v_7’s sign-index sequence is $s_7 = \langle - \quad + \quad + \rangle$. According to the definition, v_0 can be denoted by v^+; conversely, v_7 is v^-.

Such relation holds between any pair of associated v^+ and v^-:

$$v^{+}[x]-v^{-}[x]=v^{+}[x-\delta]+v^{-}[x-\delta] \qquad (5.2\text{-}1)$$

where $v[x]$ represents an element of the Hadamard vector, and x is an integer. Note that if $x<0, x>2^{n-1}-1,\ v[x] \square 0$. $\delta=2^{p-1}$, p is an integer indicating the position of the different index of a sign-index sequence, δ is a multiple of 2, $1\le\delta\le 2^{n-1}$, and n is the level of the Hadamard matrix.

Consider the example of v_0 and v_7. Since $s_0=\langle + \quad + \quad + \rangle$, $s_7=\langle - \quad + \quad + \rangle$, the only different index is the first index, $p=1, \delta=1$. So the relation can be re-written as $v^{+}[x]-v^{-}[x]=v^{+}[x-1]+v^{-}[x-1]$. The relation is explicitly explained in Figure 5.1. Simply assuming $x=2$, the relation will become $v^{+}[2]-v^{-}[2]=v^{+}[1]+v^{-}[1]$. The picture just proves this relation holds. The addition of green blocks equals subtraction between red blocks. As indicated by the arrow, changing the value of x along v^{+} and v^{-} will see the relation hold at every data point of a pair of associated Hadamard vectors.

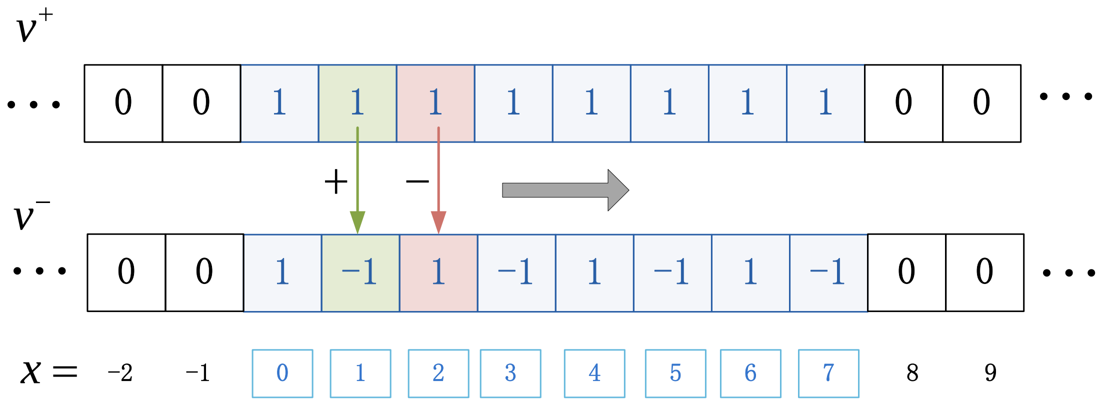


Figure 5. 1 An Example of Special Relation between Associated Vectors

Relation between Associated Hadamard Determinants

A special relation between two associated Hadamard vectors is developed. The one-dimensional case can be expanded to Hadamard determinants.

As previously mentioned, there is a one-to-one relation between a Hadamard determinant and its vector.

$$K_{z_1z_2} = v_{z_1}v_{z_2}^T \text{ or } K_{z_1z_2}\left[x, y\right] = v_{z_1}\left[x\right]v_{z_2}\left[y\right]$$

Two Hadamard determinants $K_{z_1z_2} = v_{z_1}v_{z_2}^T, K_{z_3z_4} = v_{z_3}v_{z_4}^T$, if $K_{z_1z_2}, K_{z_3z_4}$ share the same Hadamard vector $v_{z_1} = v_{z_3}$. If v_{z_2} and v_{z_4} are associated vectors, that is, $v_{z_2} = v^+$, $v_{z_4} = v^-$, then $K_{z_1z_2}, K_{z_3z_4}$ are called associated determinants. Likewise, in another direction, if $v_{z_2} = v_{z_4}$, and v_{z_1}, v_{z_3} are associated vectors, $K_{z_1z_2}, K_{z_3z_4}$ are associated determinant as well.

Definition — A pair of Hadamard determinants are defined as associated determinants if they can be factored into $K^+ = v_z\left(v^+\right)^T, K^- = v_z\left(v^-\right)^T$ or $K^+ = v^+v_z^T, K^- = v^-v_z^T$.

This is because, in one dimension, associated Hadamard vectors v^+, v^- have the relation:

$$v^+\left[x\right] - v^-\left[x\right] = v^+\left[x-\delta\right] + v^-\left[x-\delta\right]$$

There are two scenarios in a two-dimensional case.

We define the horizontal scenario if $K^+ = v_z\left(v^+\right)^T, K^- = v_z\left(v^-\right)^T$; And we define the vertical scenario if $K^+ = v^+v_z^T, K^- = v^-v_z^T$.

The one-dimensional relation can be expanded to Hadamard determinants K^+, K^-.

Horizontal scenario $K^+\left[x, y\right] - K^-\left[x, y\right] = K^+\left[x, y-\delta\right] + K^-\left[x, y-\delta\right]$

Vertical scenario $K^{+}\left[x,y\right]-K^{-}\left[x,y\right]=K^{+}\left[x-\delta,y\right]+K^{-}\left[x-\delta,y\right]$

Chapter 6: Key Ideas of new Template Matching Algorithm

Efficient Calculation of Hadamard Coefficients

This section utilizes a special relation developed in the previous chapter to achieve a fast cross-correlation calculation between a Hadamard determinant and an image block. Since cross-correlation is equivalent to the Hadamard coefficient, a highly efficient approach calculating Hadamard coefficients using special relation between associated Hadamard determinants is developed.

Cross-Correlation of Hadamard Determinant and Image Block

Cross-correlation is an approach of measuring the degree to which two parties are correlated. The standard calculation formula of cross-correlation is defined as:

$$cr_{z_1z_2} = w\square\ K_{z_1z_2} = \sum_{x=0}^{N-1}\sum_{y=0}^{N-1} w[x,y]K_{z_1z_2}[x,y] \qquad (6.1\text{-}1)$$

The value of cross-correlation is denoted by $cr_{z_1z_2}$, and it indicates the correlation degree of the spatial-domain image block w and the Hadamard determinant $K_{z_1z_2}$. It is clear that cross-correlation is obtained by the summation of corresponding items' multiplication.

Relation between Hadamard Coefficient and Cross-Correlation

As defined, the cross-correlation of a Hadamard determinant and an image block can be represented as:

$$cr_{z_1z_2} = w\square\ K_{z_1z_2} = \sum_{x=0}^{N-1}\sum_{y=0}^{N-1} w[x,y]K_{z_1z_2}[x,y]$$

Because $K_{z_1z_2}[x,y] = M[x,z_1]M[y,z_2] \Rightarrow cr_{z_1z_2} = \sum_{x=0}^{N-1}\sum_{y=0}^{N-1} w[x,y]M[x,z_1]M[y,z_2]$

And because Hadamard matrix is symmetric, $\Rightarrow M\left[x, z_1\right] = M\left[z_1, x\right]$

$$\Rightarrow cr_{z_1 z_2} = \sum_{x=0}^{N-1}\sum_{y=0}^{N-1} w\left[x, y\right] M\left[z_1, x\right] M\left[y, z_2\right] \qquad (6.1\text{-}2)$$

As mentioned, Hadamard transform coefficient $W\left[u, v\right]$ can be represented as:

$$W\left[u, v\right] = \sum_{x=0}^{N-1}\sum_{y=0}^{N-1} M\left[u, x\right] w\left[x, y\right] M\left[y, v\right]$$

$M\left[u, x\right]$ denotes the element of the Hadamard transform matrix. $w\left[x, y\right]$ is an element of the spatial-domain image block. $W\left[u, v\right]$ is the element of the converted image block in the frequency domain; it is also called the Hadamard coefficient.

By reformatting $u \to z_1, v \to z_2$, $\Rightarrow cr_{z_1 z_2} = W\left[z_1, z_2\right]$.

Thus, the cross-correlation between a Hadamard determinant and image block w is an equivalence of Hadamard coefficient obtained from the same image block and the same Hadamard determinant $K_{z_1 z_2}$.

Efficient Cross-Correlation Calculation

Recall the special relation holds between associated Hadamard determinants.

Horizontal scenario:

$$K^{+}\left[x, y\right] - K^{-}\left[x, y\right] = K^{+}\left[x, y-\delta\right] + K^{-}\left[x, y-\delta\right]$$

Vertical scenario:

$$K^{+}\left[x, y\right] - K^{-}\left[x, y\right] = K^{+}\left[x-\delta, y\right] + K^{-}\left[x-\delta, y\right]$$

To better present the relation between associated cross-correlations, we use the cross-correlation matrix $CR_{z_1 z_2}$ to represent all cross-correlation results between the entire video frame and the Hadamard determinant $K_{z_1 z_2}$.

If an image block w 's first coordinates with respect to entire frame I are $\left[x, y\right]$, we denote this image block by $w_{x,y}$. So for an $N_1 \times N_2$ frame I , $0 \le x \le N_1 - 1, 0 \le y \le N_2 - 1$. The upper-left image block is the first block of video frame I , denoted by $w_{0,0}$. An example of image block representation is shown in Figure 6.1.

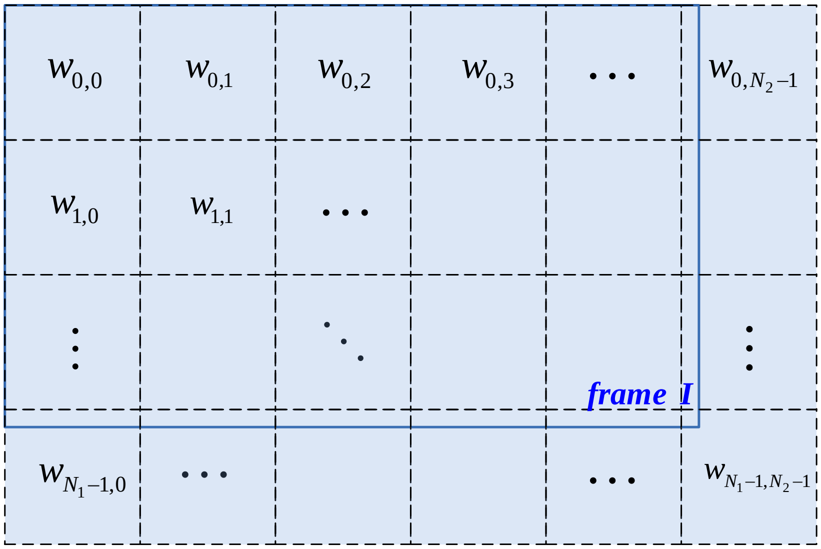


Figure 6. 1 Image Block Representation on Frame

We denote $CR^{+}\left[x, y\right] = w_{x,y} \square K^{+}$ and $CR^{-}\left[x, y\right] = w_{x,y} \square K^{-}$. $CR\left[x, y\right]$ is an element of the cross-correlation matrix.

Definition — $CR^{+}\left[x, y\right]$ and $CR^{-}\left[x, y\right]$ are called a pair of associated cross-correlations per image block $w_{x,y}$, if $CR^{+}\left[x, y\right] = w_{x,y} \square K^{+}$, $CR^{-}\left[x, y\right] = w_{x,y} \square K^{-}$, K^{+}, K^{-} are a pair of associated Hadamard determinants having special relation.

Since cross-correlation is a linear calculation procedure, the similar relation between Hadamard determinants can be found among the cross-correlation of a given image block and two associated determinants.

A special relation holds in two scenarios.

Horizontal scenario:

$$CR^{+}\left[x,y\right]-CR^{-}\left[x,y\right]=CR^{+}\left[x,y+\delta\right]+CR^{-}\left[x,y+\delta\right] \qquad (6.1\text{-}3)$$

Vertical scenario:

$$CR^{+}\left[x,y\right]-CR^{-}\left[x,y\right]=CR^{+}\left[x+\delta,y\right]+CR^{-}\left[x+\delta,y\right] \qquad (6.1\text{-}4)$$

Since associated cross-correlations have such relations,

Horizontal case:

$$CR^{+}\left[x,y\right]-CR^{-}\left[x,y\right]=CR^{+}\left[x,y+\delta\right]+CR^{-}\left[x,y+\delta\right]$$

Vertical case:

$$CR^{+}\left[x,y\right]-CR^{-}\left[x,y\right]=CR^{+}\left[x+\delta,y\right]+CR^{-}\left[x+\delta,y\right]$$

After transposition of terms, we have

Horizontal case:

$$CR^{+}\left[x,y+\delta\right]=CR^{+}\left[x,y\right]-CR^{-}\left[x,y\right]-CR^{-}\left[x,y+\delta\right] \qquad (6.1\text{-}5)$$

$$CR^{-}\left[x,y+\delta\right]=CR^{+}\left[x,y\right]-CR^{-}\left[x,y\right]-CR^{+}\left[x,y+\delta\right] \qquad (6.1\text{-}6)$$

Vertical case:

$$CR^{+}\left[x+\delta,y\right]=CR^{+}\left[x,y\right]-CR^{-}\left[x,y\right]-CR^{-}\left[x+\delta,y\right] \qquad (6.1\text{-}7)$$

$$CR^{-}\left[x+\delta,y\right]=CR^{+}\left[x,y\right]-CR^{-}\left[x,y\right]-CR^{+}\left[x+\delta,y\right] \qquad (6.1\text{-}8)$$

The equations above suggest that only two operations are required to calculate the cross-correlation for each image block with respect to a Hadamard determinant. This calculation method is very simple, and its complexity will not be affected by image block size.

Efficient Difference Measuring Criteria Calculation

To have an efficient template-matching implementation, a fast calculation of difference measuring metric must be established. New template-matching techniques use frequency-domain difference measuring metric $SATD$ (Squared Absolut Transformed Differences), $SATD$

measures the frequency-domain difference between current templates and all its possible candidate windows within the search region on the previous frame. The window with the minimum difference against the template block is selected to be the best-matched block. The best-matched block will be subtracted from the template block. The result is called a residual block, and only the residual will be transmitted through the wireless bandwidth to the receiver.

Using core ideas, a fast calculation of SATD can be generated. It is presented below:

1) Very few Hadamard coefficients applied in snake order

It is important to understand that every image block w be completely represented by a set of Hadamard determinants and corresponding coefficients, as shown in the equation:

$$w[x,y]=\frac{1}{N^2}\sum_{z_1=0}^{N-1}\sum_{z_2=0}^{N-1}W[z_1,z_2]K_{z_1z_2}[x,y]$$

Moreover, $K_{z_1z_2}$ with smaller z_1, z_2 compacts more energy of input video data into very first and very few Hadamard coefficients $W[z_1, z_2]$. As previously discussed, this frequency representation of input video can be properly utilized to calculate the difference between current template blocks and its candidate windows on the previous frame. It is more efficient to calculate frequency-domain differences measuring metric using several the very first Hadamard coefficient, and throwing out coefficients with very little energy [46].

A set of Hadamard determinants is presented in Figure 6.2. The arrow indicates the order of application. Since determinants with a smaller value z_1, z_2 have better energy compaction ability, they will be applied first. As mentioned, Hadamard dominants have excellent energy-packing ability; thus, most of energy can be compacted into first and very few Hadamard coefficients. Thus, a number of remaining coefficients can be simply discarded.

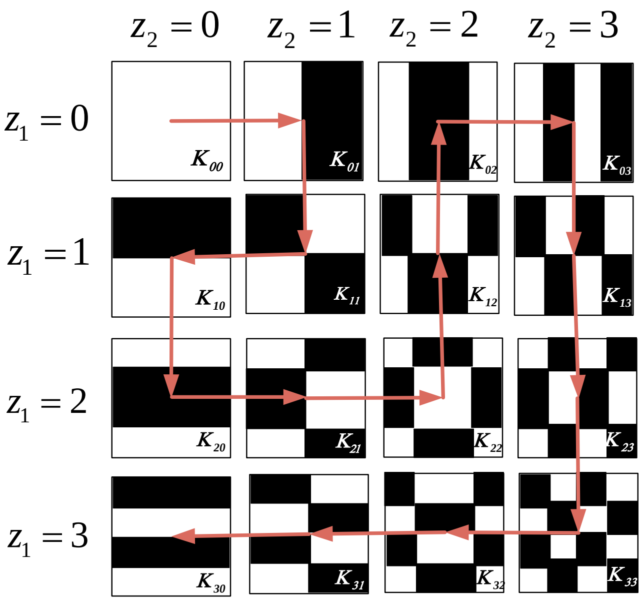


Figure 6. 2 Snake Order of Hadamard Determinant Applied

2) Equivalence relation of Hadamard coefficient and cross-correlation

As proved, the cross-correlation between Hadamard determinant $K_{z_1z_2}$ and the image block w is an equivalence of Hadamard coefficient obtained from the same image block and the same Hadamard determinant.

Cross-correlation between image block $w_{x,y}$ and Hadamard determinant $K_{z_1z_2}$ can be re-written as $CR_{z_1z_2}[x, y]$. Also, the Hadamard coefficient of the same image block $w_{x,y}$ and the same Hadamard determinant $K_{z_1z_2}$ is $W[z_1, z_2]$.

Thus, it is possible to replace Hadamard coefficients by cross-correlation and calculate SATD using a few cross-correlations.

3) A fast cross-correlation using special relation between associated Hadamard determinants

Since associated cross-correlations have such relations,

Horizontal case:

$$CR^{+}\left[x, y+\delta\right] = CR^{+}\left[x, y\right] - CR^{-}\left[x, y\right] - CR^{-}\left[x, y+\delta\right]$$

$$CR^{-}\left[x, y+\delta\right] = CR^{+}\left[x, y\right] - CR^{-}\left[x, y\right] - CR^{+}\left[x, y+\delta\right]$$

Vertical case:

$$CR^{+}\left[x+\delta, y\right] = CR^{+}\left[x, y\right] - CR^{-}\left[x, y\right] - CR^{-}\left[x+\delta, y\right]$$

$$CR^{-}\left[x+\delta, y\right] = CR^{+}\left[x, y\right] - CR^{-}\left[x, y\right] - CR^{+}\left[x+\delta, y\right]$$

Equations above suggest the calculation of cross-correlation only needs two operations.

4) Efficient calculation of *SATD*

The calculation of $SATD$ can be significantly accelerated using fast cross-correlation calculation.

Since the current template is located on the current frame, we denote the current frame by I^{j}; its previous frame is denoted by I^{j-1}. We denote current template blocks on the current frame by $t^{j}_{x_1,y_1}$ and denote all possible candidate windows on the previous frame by $w^{j-1}_{x_2,y_2}$.

$CR^{j}_{z_1z_2}\left[x_1, y_1\right] = t^{j}_{x_1,y_1} \square\ K_{z_1z_2}$ represents cross-correlation between current template $t^{j}_{x_1,y_1}$ and Hadamard determinant $K_{z_1z_2}$, $CR^{j-1}_{z_1z_2}\left[x_2, y_2\right] = w^{j-1}_{x_2,y_2} \square\ K_{z_1z_2}$ denotes cross-correlation between current template $w^{j-1}_{x_2,y_2}$ and Hadamard determinant is denoted by $K_{z_1z_2}$. So, $SATD$ can be calculated by:

$$SATD = \sum_{z_1=0}^{m_1}\sum_{z_2=0}^{m_2}\left|CR^{j}_{z_1z_2}\left[x_1, y_1\right] - CR^{j-1}_{z_1z_2}\left[x_2, y_2\right]\right| \qquad (6.2\text{-}1)$$

Because most energy is packed into very few cross-correlations within smaller z_1, z_2, the value of m_1, m_2 can be very small. Simulation suggests $m_1, m_2 = 2$ can create a sufficient performance. Thus, the calculation of $SATD$ only requires the summation of six absolute differences. According to the proposed snake order of Hadamard determinants, all required coordinates of cross-correlations only include $[m_1, m_2] = [0,0],[0,1],[1,1],[1,0],[2,0],[2,1]$.

New Template Matching Algorithm

Using core ideas of efficient cross-correlation, a new template-matching algorithm is established. As mentioned previously, there are three types of requirements brought by a wide variety of construction applications. Those requirements are 'timely' and 'precise'. Since different construction applications have different requirements or requirements in different priorities, the parameters can be flexibly adjusted to meet requirements in terms of different scenarios.

Table 6. 1 Relation between Parameters and Requirements

Requirement	Parameter	Approach
Precise	m_1, m_2 in frequency domain m_3 in special domain	Larger value of parameters generates higher accuracy
Timely	m_1, m_2, m_3	Smaller value of parameters results in more quickly implementation

According to the snake order of implementation, Table 6.2 describes the number of Hadamard determinants applied in total and $[m_1, m_2]$, which denotes the last coordinates of the applied Hadamard determinant.

Table 6. 2 Relation between Total amount of Hadamard Determinants and Value of m_1, m_2

Amount of Hadamard determinants	Hadamard determinants applied	$[m_1, m_2]$
1	K_{00}	$[0,0]$
2	K_{00}, K_{01}	$[0,1]$
3	K_{00}, K_{01}, K_{11}	$[1,1]$

4	$K_{00}, K_{01}, K_{11}, K_{10}$	$[1,0]$
5	$K_{00}, K_{01}, K_{11}, K_{10}, K_{20}$	$[2,0]$
6	$K_{00}, K_{01}, K_{11}, K_{10}, K_{20}, K_{21}$	$[2,1]$
7	$K_{00}, K_{01}, K_{11}, K_{10}, K_{20}, K_{21}, K_{22}$	$[2,2]$
8	$K_{00}, K_{01}, K_{11}, K_{10}, K_{20}, K_{21}, K_{22}, K_{12}$	$[1,2]$
9	$K_{00}, K_{01}, K_{11}, K_{10}, K_{20}, K_{21}, K_{22}, K_{12}, K_{02}$	$[0,2]$
10	$K_{00}, K_{01}, K_{11}, K_{10}, K_{20}, K_{21}, K_{22}, K_{12}, K_{02}, K_{03}$	$[0,3]$

A specific procedure of efficient template matching algorithm is provided.

For each current frame I^j *:*

Step i *Pre-define* $\left[m_1, m_2\right]$ *, which denotes the last coordinates of snake-ordered Hadamard determinants used for efficient calculation of cross-correlations.*

Step ii *Calculate cross-correlation of entire current frame* I^j *and the pre-defined Hadamard determinants* $K_{z_1z_2}$ *in an efficient manner using the special relation between associated Hadamard determinants. The result is frame-size cross-correlation matrix. Store the results in memory.*

Step iii *For each current template block* $t^j_{x_1,y_1}$ *:*

1) Calculating the SATD between current template $t^j_{x_1,y_1}$ *and its candidate windows* $w^{j-1}_{x_1,y_2}$ *using stored cross-correlations.*

$$SATD = \sum_{z_1=0}^{m_1} \sum_{z_2=0}^{m_2} \left| CR^j_{z_1z_2}\left[x_1, y_1\right] - CR^{j-1}_{z_1z_2}\left[x_2, y_2\right] \right|$$

2) Select m_3 *windows with minimum SATD, calculate SAD between current template block and remaining windows.*

3) Of m_3 *remaining windows, select the window with the minimum SAD as best-matched block.*

The new template-matching algorithm has two selections to find the best-matched window of current template block.

- The first round selection uses a special relation between associated Hadamard determinants, and the special relation reduces computational complexity. It requires two subtractions to calculate the cross-correlation of the current template and all its candidate windows. It is important that the search region of the proposed algorithm contains the entire video frame. The proposed algorithm has the same amount of candidate windows as the exhaustive search technique; that is, all possible image blocks are involved in the matching procedure. The first selection uses some Hadamard determinants indicated by $[m_1, m_2]$ to calculate matching criteria $SATD$, and it selects m_3 remaining windows with minimum $SATD$.
- Among m_3 windows, the second selection will calculate spatial-domain matching criteria SAD, and it will choose the window with the smallest SAD as the best-matched block of the current template block.

Chapter 7: Simulation Results

Simulation Results

The proposed efficient template-matching algorithm is compared with exhaustive search, three-step search, four-step search and hexagon search. Implementation was performed on the same hardware configuration Pentium 4 PC at 2.5GHz running Windows 7.

The proposed algorithm makes use of three parameters m_1, m_2, m_3 to lever the key performance index (KPI) of the template-matching technique. Table 7.1 displays the relation between parameters and the KPI. By adjusting the value of m_1, m_2, m_3, KPI will be affected.

Table 7. 1 Relation between Parameters, Requirements and KPI

Parameter	Requirement	Key performance index
m_1, m_2, m_3	Precise	$PSNR$
m_1, m_2, m_3	Timely	$runtime$

To better interpret the simulation results, three sections are set for first two KPI, including $PSNR$ and $runtime$, and an analysis of computational complexity will also be provided.
The following popular test-videos are used. The videos are shown in a decreasing order of motion content amount. The first several videos contain a large amount of movements. The last few videos are very stationary.

Table 7. 2 List of Test-videos

Video sequence	Format	Image size
Table tennis	SIF	352×240
Football	SIF	352×240
Football	CIF	352×288
Stefan	QCIF	176×144
Stefan	CIF	352×288
Foreman	QCIF	176×144
Foreman	CIF	352×288
Coastguard	QCIF	176×144
Car phone	CIF	352×288
Container	CIF	352×288
Hall monitor	CIF	352×288
Silence	CIF	352×288
News	CIF	352×288
Trevor	QCIF	176×144
Flower garden	SIF	352×240
Flower garden	CIF	352×288
Mobile calendar	SIF	352×240
Mobile calendar	CIF	352×288
Grandma	QCIF	176×144
Miss america	QCIF	176×144
Mother and daughter	QCIF	176×144
Salesman	QCIF	176×144
Akiyo	QCIF	176×144
Akiyo	CIF	352×288

Precision Measurement Results

The accuracy of the template-matching algorithm is usually measured by average SAD per template and $PSNR$. The average SAD per template is first obtained from calculating a summation of SAD between all original templates on all frames along video sequence and their best-matched blocks on corresponding previous frames, and then dividing this number by total amount of templates of video sequence. $PSNR$ is Peak-Signal-to-Noise-Ratio function, defined as:

$$PSNR = 10\log_{10}\frac{R^2}{MSE} \qquad (7.1\text{-}1)$$

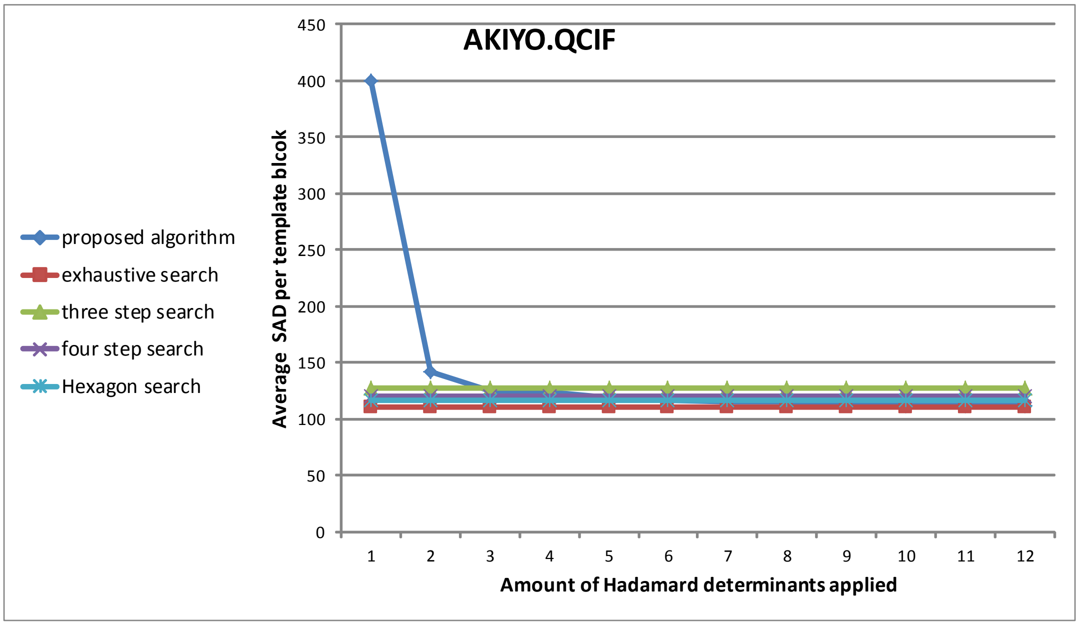
where R denotes the maximum fluctuation of input data. In this case R equals the summation of every pixel's peak value of entire frame. For example, if the input frame is size 352×288, and each pixel is represented by 8 bits, $R = 352 \times 288 \times 2^8$.

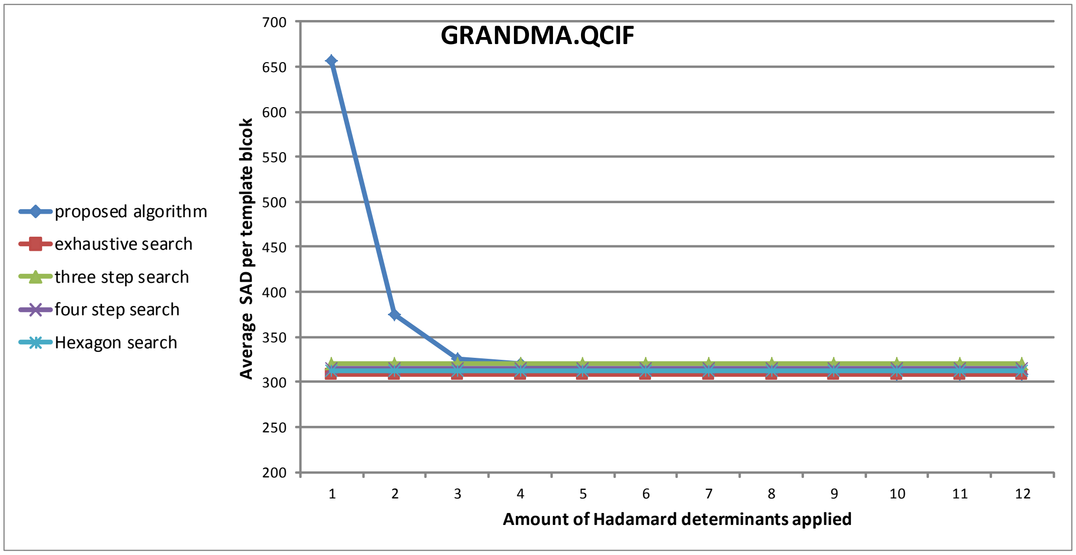
MSE is the mean squared error, defined as

$MSE = \frac{1}{N_1 N_2}\sum_{x=1}^{N_1}\sum_{y=1}^{N_2}\left(I^j[x,y] - I^{j-1}[x,y]\right)^2$, $N_1 \times N_2$ is the size of video frame. It is clear that a higher $PSNR$ indicates high accuracy of template matching.

Accuracy performance is first measured with the average SAD per template, and four videos in table are chosen to serve as test-videos.

Figure 7.1 shows the performance in terms of varying $[m_1, m_2]$ (denoted by 'amount of Hadamard determinants') with a constant value of m_3. In this case, we keep $m_3 = 5$. Four other popular template-matching algorithms, including exhaustive search, three-step search, four-step search and hexagon search are used to compare to the new algorithm. The template-matching accuracy is measured by the average SAD per template (dividing total video sequence SAD by product of number of frames per video and number of templates per frame). The results, indicated by an increasing number of Hadamard determinants, will result in higher accuracy.


AKIYO.QCIF
Average SAD per template blcok
0
50
100
150
200
250
300
350
400
450
1
2
3
4
5
6
7
8
9
10
11
12
Amount of Hadamard determinants applied
proposed algorithm
exhaustive search
three step search
four step search
Hexagon search


GRANDMA.QCIF
Average SAD per template blcok
200
250
300
350
400
450
500
550
600
650
700
1
2
3
4
5
6
7
8
9
10
11
12
Amount of Hadamard determinants applied
proposed algorithm
exhaustive search
three step search
four step search
Hexagon search

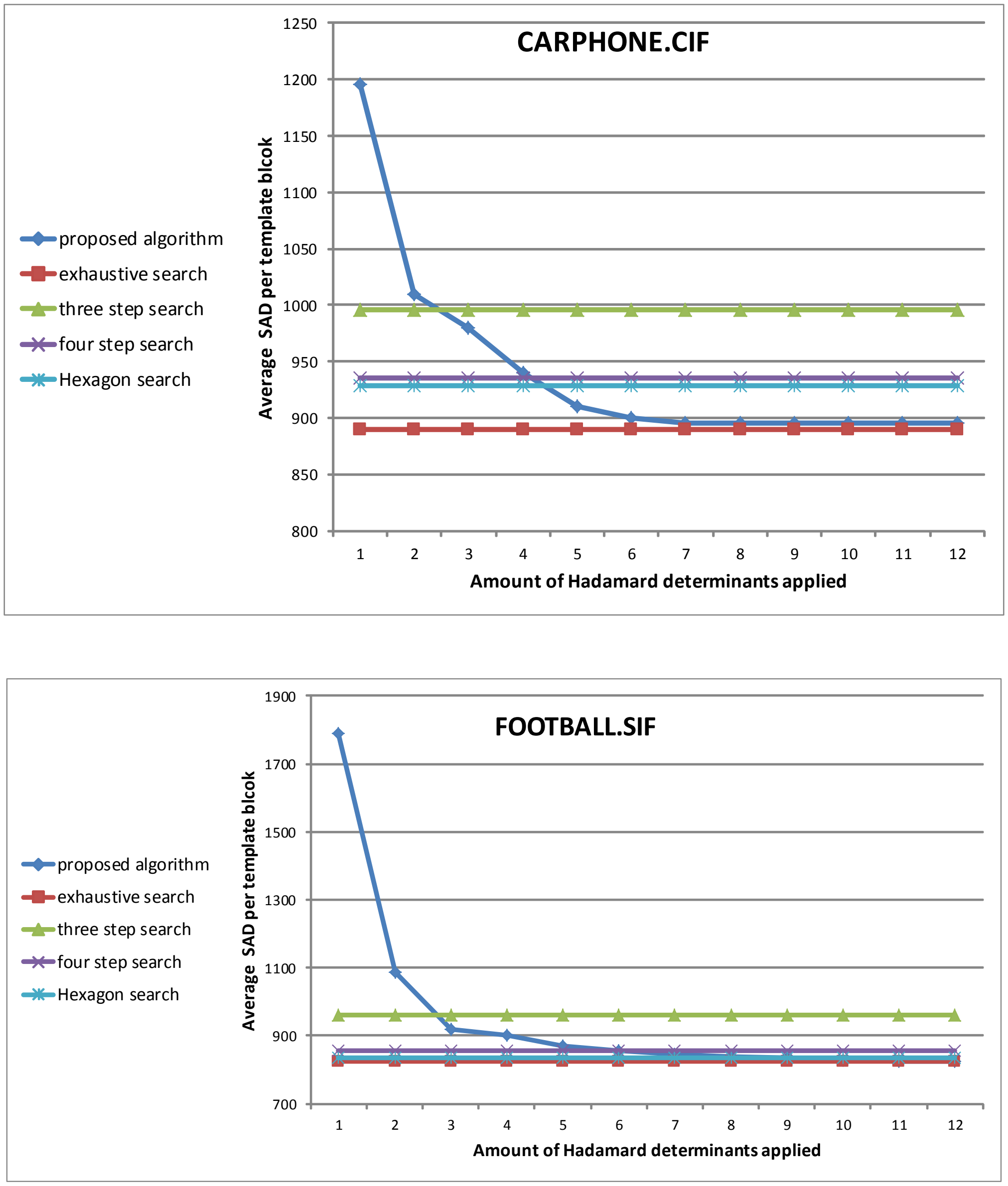
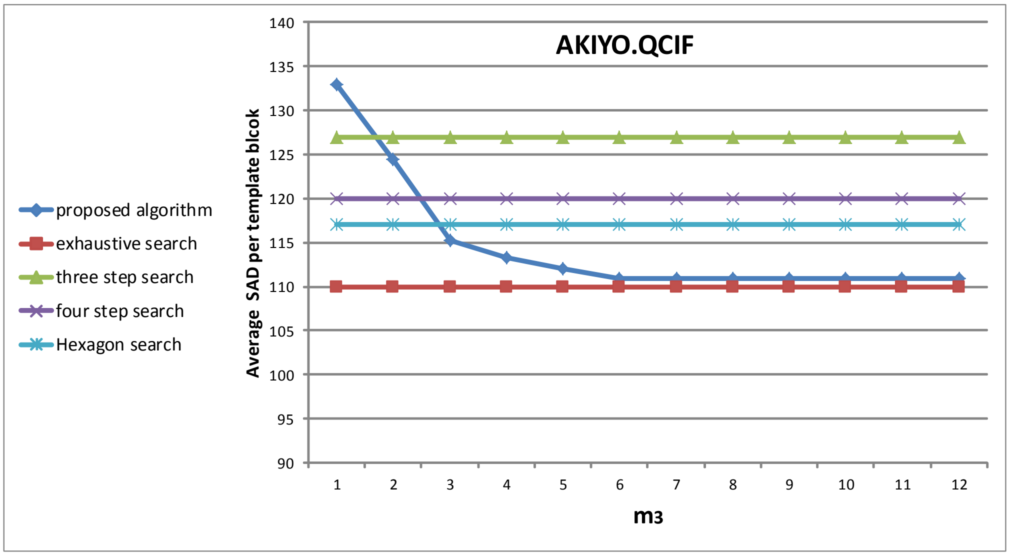


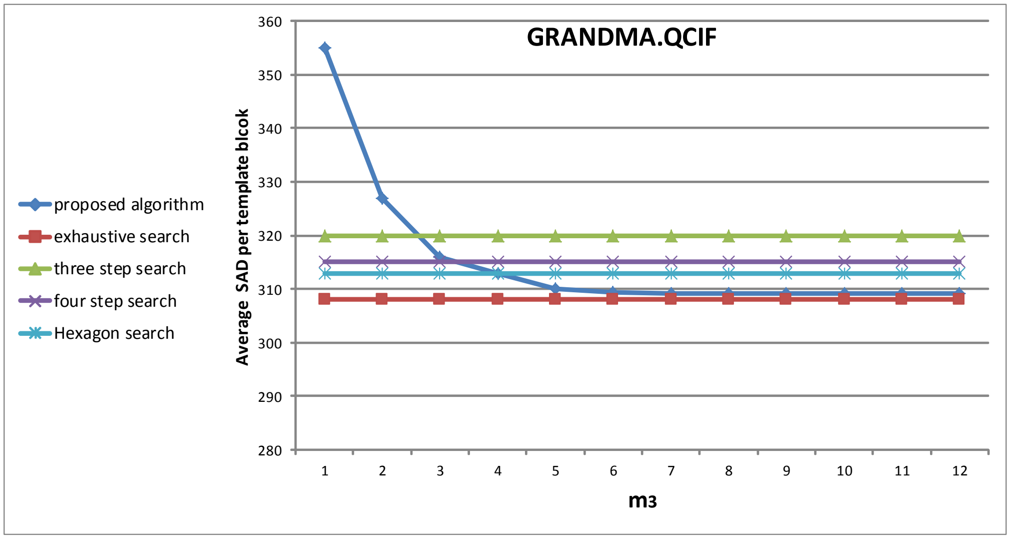
Figure 7. 1 Comparison of Average SAD per Template Block with $m_3 = 5$

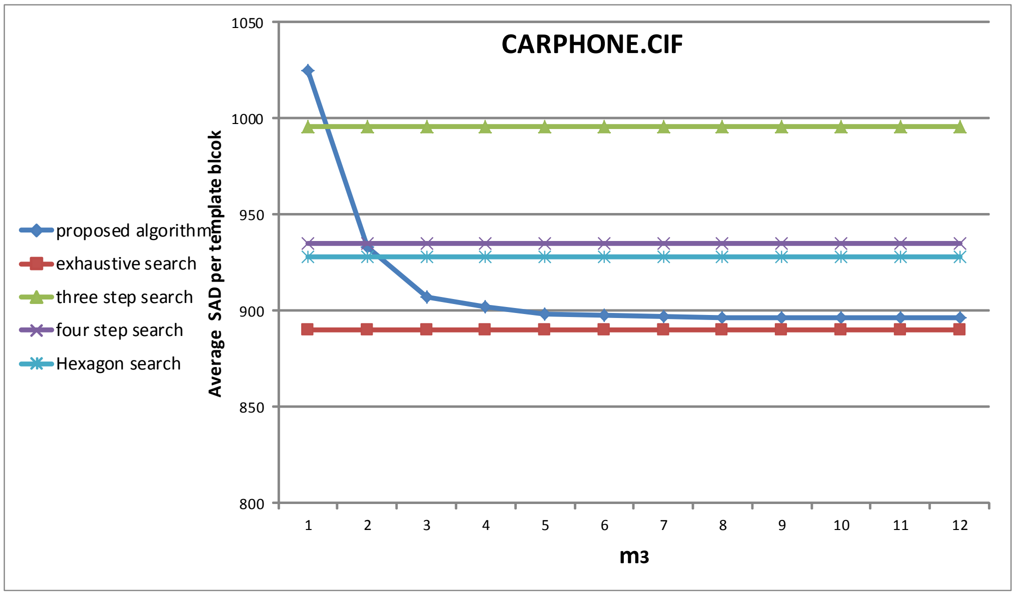
For all video streams, it is clear that proposed algorithm outperforms three-step search for applying 3 Hadamard determinants. For all video streams except one, the proposed algorithm outperforms four step search and hexagon search for using 4 Hadamard determinants, and for Football.sif, the proposed algorithm requires 5 Hadamard determinants to outperform four step search, and it requires 6 determinants to outperform hexagon search. It is obvious that Football.sif has a lower convergence velocity than Akiyo.qcif. This is because compared with Football.sif, Akiyo.qcif is a static video sequence so its templates can be matched using smaller number of Hadamard determinates.

Similarly, it is also possible to maintain the number of Hadamard determinants as a constant, and select different value of m_3, where Figure 7.2 describes accuracy with respect to varying m_3, with the constant value of $[m_1, m_2] = [2, 0]$, in which case, 5 Hadamard determinants are used to compact energy of input data.

The results prove higher value of m_3 will create a more precise performance. And results of new algorithm converge to the optimal, which means increasing the number of residual windows for which spatial SAD is calculated generates lower SAD values, and the SAD values of proposed algorithm can approach those of exhaustive search. All video streams results suggest that proposed algorithm with $m_3 = 3$ is able to outperform three step search, with $m_3 = 4$ is able to outperform four step search, and with $m_3 = 5$ is able to outperform hexagon search.


AKIYO.QCIF
Average SAD per template blcok
140
135
130
125
120
115
110
105
100
95
90
1
2
3
4
5
6
7
8
9
10
11
12
m3
proposed algorithm
exhaustive search
three step search
four step search
Hexagon search


GRANDMA.QCIF
Average SAD per template blcok
360
350
340
330
320
310
300
290
280
1
2
3
4
5
6
7
8
9
10
11
12
m3
proposed algorithm
exhaustive search
three step search
four step search
Hexagon search



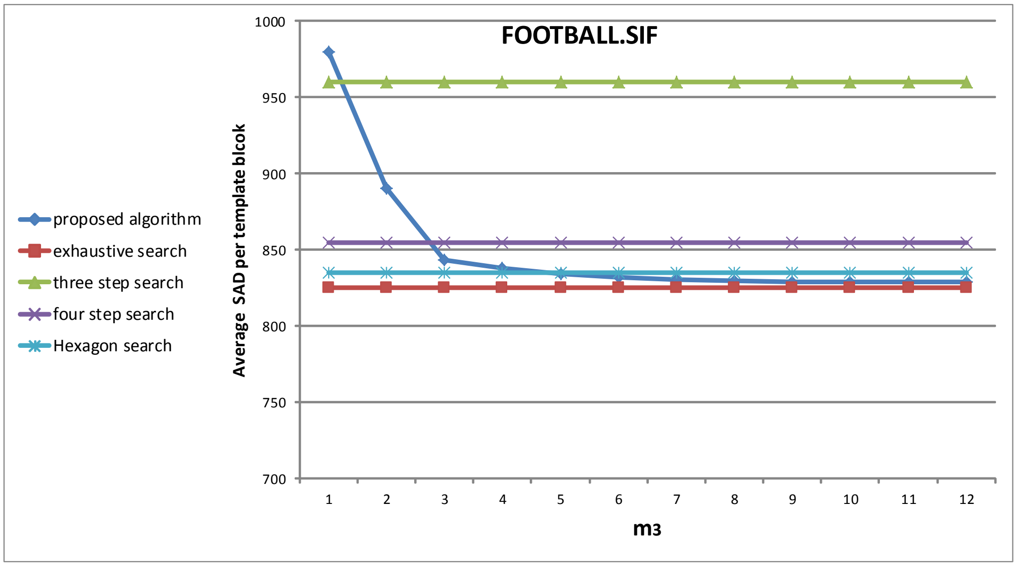


Figure 7. 2 Comparison of Average SAD per Template Block with $[m_1, m_2] = [2, 0]$

Simulation results suggest that by increasing the value of three parameters m_1, m_2, m_3, the algorithm will get higher performance accuracy. Moreover, the proposed template-matching algorithm is able to converge to the highest accuracy boundary of exhaustive search as a very high speed. It outperforms three-step search, four-step search and hexagon search without enlarging parameters to large integers.

Additionally, a table that specifically compares $PSNR$ in terms of proposed algorithms and other popular algorithms is presented. The proposed algorithm uses constant parameters, $[m_1, m_2] = [2,1], m_3 = 5$. Six Hadamard determinants are applied in the first selection, and five windows are remained in second selection. The results involve 24 test videos.

Table 7. 3 Performance Comparison of $PSNR(dB)$

Video sequence	Format	Image size	$PSNR(dB)$				
			Proposed algorithm	**Exhaustive search**	**Three step search**	**Four step search**	**Hexagon search**
Table tennis	SIF	352×240	26.61	27.49	25.85	26.44	26.49
Football	SIF	352×240	22.83	23.11	21.67	22.27	22.30
Football	CIF	352×288	25.35	25.93	24.61	24.66	25.12
Stefan	QCIF	176×144	23.00	23.58	21.98	22.47	22.49
Stefan	CIF	352×288	25.23	25.86	22.43	23.93	24.66
Foreman	QCIF	176×144	30.23	30.86	30.18	29.97	29.77
Foreman	CIF	352×288	31.58	32.58	30.68	27.42	27.49
Coastguard	QCIF	176×144	33.21	33.69	32.85	32.97	33.04
Car phone	CIF	352×288	31.99	32.34	31.07	31.84	31.89
Container	CIF	352×288	38.44	38.49	37.54	38.25	38.27
Hall monitor	CIF	352×288	39.09	39.59	38.44	39.00	39.03
Silence	CIF	352×288	35.92	36.25	35.26	35.57	35.89
News	CIF	352×288	36.85	37.02	36.22	36.57	36.74
Trevor	QCIF	176×144	37.99	38.22	37.03	37.83	37.96
Flower garden	SIF	352×240	23.86	25.28	22.58	23.04	23.12
Flower garden	CIF	352×288	25.53	25.68	25.50	25.52	25.52
Mobile calendar	SIF	352×240	22.61	22.67	22.33	22.57	22.59
Mobile calendar	CIF	352×288	23.67	23.89	23.04	23.74	23.84

Grandma	QCIF	176×144	37.03	37.94	36.49	36.90	36.90
Miss america	QCIF	176×144	26.61	26.99	25.45	26.57	26.61
Mother and daughter	QCIF	176×144	39.67	39.78	39.59	39.63	39.68
Salesman	QCIF	176×144	40.06	40.55	39.02	39.75	39.78
Akiyo	QCIF	176×144	45.13	44.88	44.21	44.99	45.13
Akiyo	CIF	352×288	43.24	43.24	43.06	43.12	43.24

According to the above comparisons, the general conclusion is that the performance in terms of accuracy of proposed algorithm is better than a three-step search, a four-step search and a hexagon search. It can even converge to the performance of exhaustive search in some cases, namely, increasing the amount of Hadamard determinants and the amount of remaining candidate windows generates lower SAD values. In both SAD and $PSNR$ cases, the proposed algorithm approaches values of exhaustive search.

Runtime Measurement Results

The following figure illustrates the runtime of proposed algorithm and exhaustive search, three-step search, four-step search and hexagon search. We use one of the highest-motion video sequence, $\text{Stefan.QCIF}\left(176\times 144\right)$, to be test video. The proposed algorithm uses constant parameters, $\left[m_1, m_2\right]=\left[2,1\right], m_3=5$.

Also, to have better comparison, we enlarge the size of each frame of the video sequence by padding more pixels. The y-axis represents runtime measured by second, and the x-axis represents frame size in increasing order.

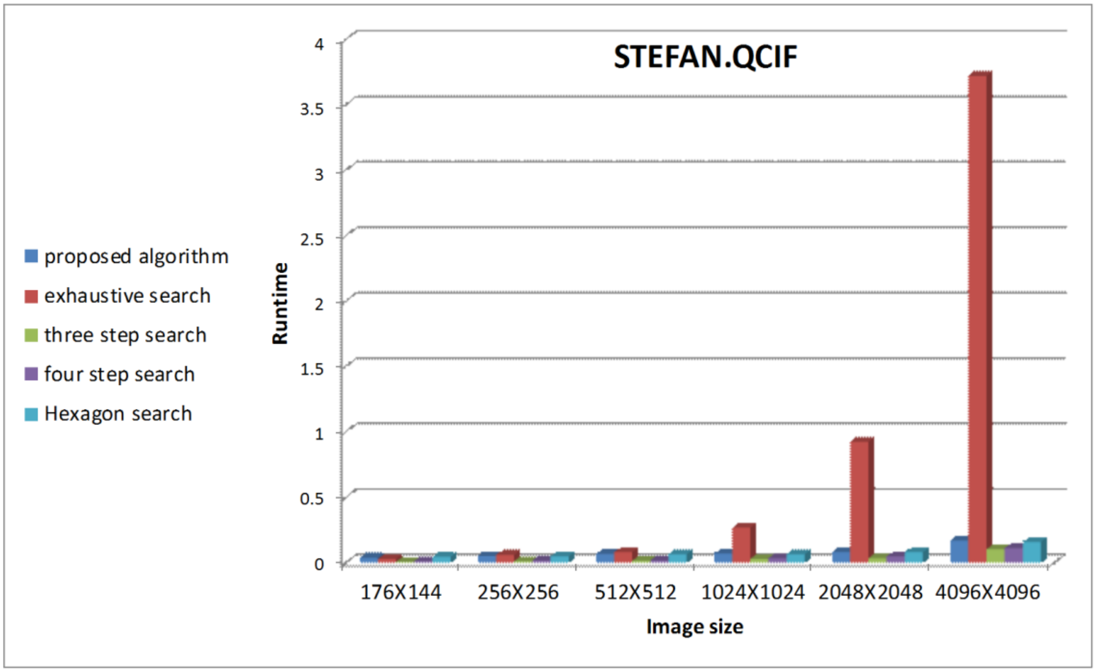


Figure 7. 3 Runtime Comparison

The proposed algorithm has same level of runtime as popular fast template-matching algorithms, such as three step search, four step search and hexagon search. This makes it more applicable in real-time transmissions.

Analysis of Computational Complexity

The three parameters m_1, m_2 and m_3 affect both memory and time complexity. To be clear, m_1, m_2 represent the indices of Hadamard determinants used for calculating Hadamard coefficients for each image block, and m_3 represents the amount of residual candidate windows for which the special SAD is calculated.

As to memory complexity, as shown in the algorithm description, for each frame, there are $N_1 \times N_2$ candidate windows in all. And for each candidate windows, there are $(m_1 + 1)(m_2 + 1)$ Hadamard coefficients need to be stored in memory. And both current frame and previous frame need to be considered for each template matching procedure. In total, the memory complexity is $2(m_1+1)(m_2+1)N_1N_2$, where m_3 do not affect the memory complexity.

As to temporal complexity, we assume frame size is $N_1 \times N_2$ and image block size is $N \times N$, and assume 1 time unit for each operation involving addition, subtraction, multiplication, absolute value and taking minimum of two numbers. As shown in Figure 7.4, every frame has $(m_1+1)(m_2+1)$ corresponding Hadamard-domain projection frames, which can be called Hadamard-coefficients frames. And except for the first coefficient frame, any other coefficient frame can be calculated using the special relation between adjacent Hadamard determinants.

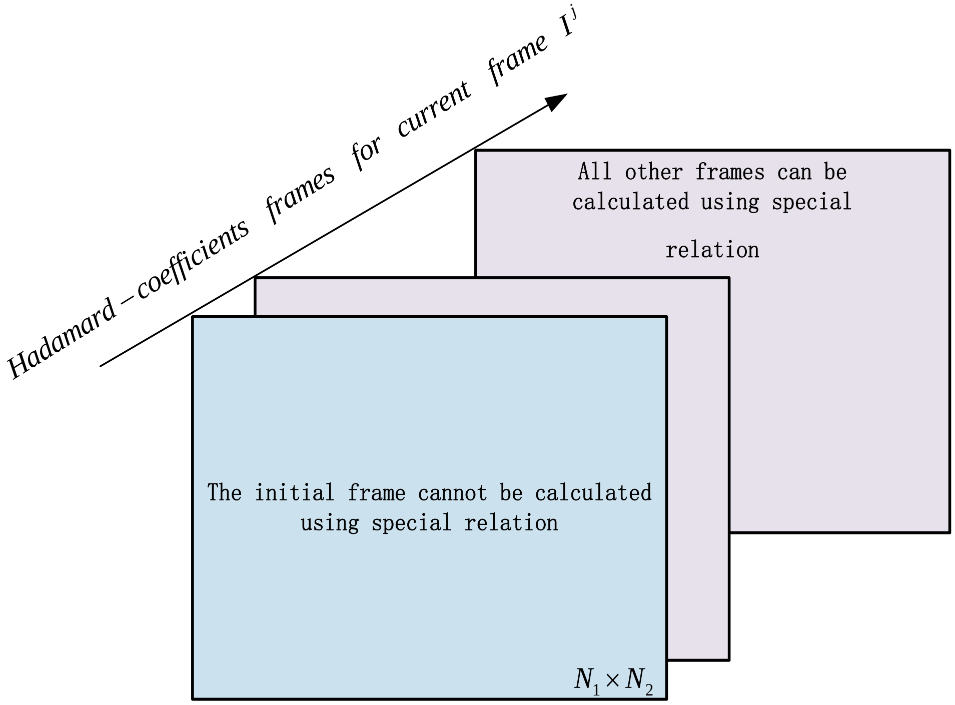

$Hadamard-coefficients\ \ frames\ \ for\ \ current\ \ frame\ \ I^j$
All other frames can be calculated using special relation
The initial frame cannot be calculated using special relation
$N_1 \times N_2$

Figure 7. 4 Hadaamrd-coefficients Frames

Table 7.4 shows the analysis of temporal complexity per frame.

Table 7. 4 Analysis of Temporal Complexity

Implementation per frame (Including all templates in frame)	**Temporal complexity**	**Complexity level**

Calculate first Hadamard-coefficients frame	$k_1N_1N_2$ (k_1 is a constant number)	$k_1N_1N_2$
Calculate other Hadamard-coefficients frames using special relation	$2\left[(m_1+1)(m_2+1)-1\right]N_1N_2$	$k_2N_1N_2$
Calculate SATD	$\left[3(m_1+1)(m_2+1)-1\right]N_1N_2\times\frac{N_1N_2}{N^2}$	$k_3N_1N_2$
Select m_3 **minimum SATD**	$(m_3-1)N_1N_2$	$k_4N_1N_2$
Calculate m_3 **real spatial domain SAD and select the minimum one**	$\left[(3N^2-1)m_3+(m_3-1)\right]\times\frac{N_1N_2}{N^2}$	$k_5N_1N_2$

Hence, the temporal complexity is $O(kN_1N_2)$ per video frame. And temporal complexity for proposed algorithm is similar as those for fast search algorithms like four step search ($O(kN_1N_2)$), and is remarkably less than exhaustive search ($O(N_1^2N_2^2)$).

Chapter 8: Summary and Conclusions

Summary

The proposed efficient template-matching algorithm is compared with most widely used methods including exhaustive search, three-step search, four-step search and hexagon search. The results show proposed algorithm is beneficial for construction projects since it deliver a better accuracy than most methods. Moreover, it approaches to the optimal performance by only consuming limited computational resources, so it is able to process jobsite video streams in real time. Also, the proposed algorithm can adjust to multiple requirements during different stages of construction projects by properly adjusting three parameters.

Conclusions

Template matching usually consumes the most computation in video processing systems, whereas commonly used template-matching techniques are unable to meet various requirements of various construction project applications. A new template-matching technique in the frequency domain presented in this dissertation can be adjusted to meet various and varying requirements.

The proposed algorithm makes use of Hadamard determinants' energy compaction ability to approximate spatial-domain difference measuring criteria (matching criteria) using very few yet representative Hadamard coefficients in the frequency domain. Additionally, it uses equivalence between cross-correlation and Hadamard coefficient to develop a substitute-matching criteria using cross-correlation for every matching attempt. A special relation between two associated Hadamard determinants is explored to accelerate computation of cross-correlations by requiring only two operations per template per determinant. Additionally, the proposed algorithm utilizes three parameters to adjust performance, such as precision and timeliness flexibility. Thus it is superior for various construction-related applications.

Simulation results show the proposed algorithm outperforms many common template-matching algorithms such as three step search without largely increasing the computational complexity. Additionally, it approaches to SAD values (which measure accuracy) of exhaustive search if increasing the parameters accordingly. But the computational complexity of proposed algorithm is similar as that of popular algorithms such as three step search, four step search and hexagon search. This is because the proposed algorithm only accelerates the computational implementation of matching process without eliminating the amount of candidate windows locating on reference frame.

Bibliography

[1] F. Dufaux and F. Moscheni. (1995) “Motion estimation techniquesfor digital TV – a review and a new contribution”, Proc.IEEE 83, 858–876.

[2] M. Abderrahim, E. Garcia. (2005) R. Diez and C. Balahuer, A Mechatronics Security System for the Construction Site, Automation in Construction , 460-466.

[3] http://www.bls.gov/iif/.

[4] https://www.osha.gov/Publications/OSHA3252/3252.html.

[5] Ung Kyun Lee, Joo Heon Kim. (2009) Development of a Mobile Safety Monitoring System for Construction Sites, Automation in Construction, 258-264.

[6] J. B. Lee and H. Kalva. (2008)The VC−1 and H.264 Video Compression Standards for Broadband Video Services, Springer Science Business Media, New York.

[7] David Hillman. (1999) Multimedia Technology & Applications, Delmar Publishers.

[8] R. C. Gonzalez, R. E. Woods, Digital Image Processing, Third Edition.

[9] Ahmed, N., Natarajan, T., Rao, K.R. (1974) "Discrete Cosine Transform", IEEE Transactions on Computers C–23 (1): 90–93.

[10] Brigham, E. Oran. (1988) The fast Fourier transform and its applications, N.J.: Prentice Hall.

[11] Townsend, W. J.; Thornton, M. A. Walsh Spectrum Calculations Using Cayley Graphs.

[12] Kunz, H.O. (1979) On the Equivalence Between One-Dimensional Discrete Walsh-Hadamard and Multidimensional Discrete Fourier Transforms. IEEE Transactions on Computers 28 (3): 267–8.

[13] Smith, Steven W. (1999) Chapter 8: The Discrete Fourier Transform. The Scientist and Engineer's Guide to Digital Signal Processing (Second ed.). California Technical Publishing.

[14] I.E.G. Richardson. (2002) Video Codec Design: Developing Image and Video Compression Systems, edited by John Wiley & Sons, Chichester.

[15] R. Brunelli. (2009) Template Matching Techniques in Computer Vision: Theory and Practice, Wiley.

[16] T. Moon and W. Stirling. (2000) Mathematical Methods and Algorithms for Signal Processing. Prentice Hall.

[17] Korman S., Reichman D., Tsur G. and Avidan S. (2013) "FAsT-Match: Fast Affine Template Matching", CVPR.

[18] H.264 and MPEG-4 Video Compression. (2003) Video Coding for Next-generation Multimedia. Chichester: John Wiley & Sons Ltd. E. G. Richardson, Iain.

[19] An introduction to the Theory of Statistics , McGraw Hill, Mood & Graybill.

[20] Borko Furht, Joshua Greenburg and Raymond Westwater. (1997) Motion Estimation Algorithms for Video Compensation, Kluwer Academic Publishers.

[21] M. Brunig and W. Niehsen. (2001) Fast full−search block matching, IEEE Trans. Circ. Syst. Vol. 11, 241–247.

[22] Xie L. Y., Su X. Q., Zhang S. (2010) A Review of Motion Estimation Algorithms for Video Compression.

[23] S. Zafar, Y.Q. Zhang and J. S. Baras. (1991) Predictive Block-Matching Motion Estimation for TV coding---Part I: Inter-Block Prediction, IEEE Trans. On Broadcasting, Vol.37, No.3, pp.97-101.

[24] B. Liu and A. Zaca. (1993) New Fast Algorithms for the Estimation of Block Motion Vectors, IEEE Trans. on Circuits and Systems for Video Technology, Vol.3, No.2, pp.148-157.

[25] P.L. Lai and A. Ortega. (2006) Predictive fast motion/disparity search for multiview video coding, Proc. IEEE Visual Communications and Image Processing 6077, pp. 7709 ‒ 7709, San Jose.

[26] Eric Chan. (1993) Review of Block Matching Based Motion Estimation Algorithms for Video Compression, IEEE Trans. pp.151-153.

[27] H. Jelveh and A. Nandi. (1991) Improved variable size block matching motion compensation for video conferencing applications, Digital Signal Processing, A. Cappellini and A. Constantinides, Eds. Berlin, Germany: Springer-Verlag.

[28] I. Rhee, G. R. Martin. (2000) Muthukrishnan and R. A. Packwood, Quad tree-Structured Variable-Size Block- Matching Motion Estimation with Minimal Error, IEEE Trans. on Circuits and Systems for Video Technology,Vol. 10,pp. 42-50.

[29] Ishfaq Ahmad, Weiguo Zheng, Jiancong Luo, Ming Liou. (2006) A Fast Adaptive Motion Estimation Algorithm, IEEE Transactions on Circuits And Systems For Video Technology, Vol. 16, No. 3, pp.420-438.

[30] Xuan Jing, Lap-Pui Chau. (2004) An Efficient Three-Step Search Algorithm for Block Motion Estimation, IEEE transactions on multimedia: 435-437.

[31] Chen Lu; Wang. (2010) Diamond Search Algorithm, ECE, University of Texas.

[32] Bhavina Patel, R.V.Kshirsagar, Vilas Nitnaware, Sarvajanik, Surat, and Nagpur, review and comparative study of motion estimation techniques to reduce complexity in video compression.

[33] Lai Man Po and Wing Chung Ma. (1996) A Novel Four-Step Search Algorithm for Fast Block Motion Estimation, Circuits and Systems for Video Technology, IEEE Trans. Vo.6, No.3, pp.313-317.

[34] L. Baumert, S. W. Golomb and M. Hall, Jr. (1962) Discovery of an Hadamard matrix of order 92, Bull. Am. Math. Soc., vol. 68, pp. 237-238.

[35] Paley, R. E. A. C. (1993) On orthogonal matrices, Journal of Mathematics and Physics 12: 311–320.

[36] S. Omachiand and M. Omachi. 2007. Fast template matching with polynomials, IEEE Trans. Image Process., vol. 16, no. 8, pp.2139 – 2149.

[37] Williamson, J. (1944) Hadamard’s determinant theorem and the sum of four squares, Duke Mathematical Journal 11 (1): 65–81.

[38] Wanless, I.M. (2005) "Permanents of matrices of signed ones". Linear and Multilinear Algebra 53: 427–433.

[39] C. M. Mak., C. K. Fong and W. K. Cham. (2010). Fast motion estimation for H.264/AVC in Walsh Hadamard domain, IEEE Trans. Circuits Syst. Video Technol.

[40] N. J. Fine. (1949) On the Walsh functions, Trans. Am. Math. Soc., vol. 65, pp. 372414.

[41] Ritter, Terry. (1996) Walsh-Hadamard Transforms: A Literature Survey.

[42] C. K. Yuen. (1972) Remarks on the Ordering of Walsh Functions, IEEE Transactions on Computers 21(12): 1452.

[43] Y. Moshe, H. Hel-Or. (2006) A Fast Block Motion Estimation Algorithm Using Gray Code Determinants, Proc. IEEE Symp. Signal Processing and Information Technology, pp.185 - 190, Vancouver, Canada.

[44] Horn, Roger A., Johnson, Charles R. (1991) Topics in Matrix Analysis, Cambridge University Press.

[45] Steeb, Willi-Hans. 1997. Matrix Calculus and Kronecker Product with Applications and C++ Programs, World Scientific Publishing.

[46] H. Kitajima. (1976) Energy Packing Efficiency of the Hadamard Transform, *IEEE Trans. Commun.,* vol. 24, no. 11, pp. 1256-1258.