COMPUTER VISION FOR RAILROAD TRACK INSPECTION

BY

ESTHER INEZ RESENDIZ

DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Electrical and Computer Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2010

Urbana, Illinois

Doctoral Committee:

Professor Narendra Ahuja, Chair
Professor Christopher P. L. Barkan
Professor Thomas S. Huang
Professor Douglas L. Jones

ABSTRACT

In railroad track inspection, the inspection images contain periodically occurring components. Computer vision has recently been applied to several railroad applications due to its potential to improve the efficiency, objectivity, and accuracy when analyzing large databases of acquired video and images. We utilize those promising results to develop a more general method to detect and segment any periodically occurring objects in an image. The techniques used to analyze the periodically occurring track components could be used to analyze a broader class of images which contain periodically repeating objects that are similar, but not identical. We demonstrate how spectral estimation-based methods can be used to extract periodically repeating components in track inspection video.

Periodically occurring activities occur in many videos. Particularly in biological applications, activities tend to be formed from one or two characteristic poses that move in a repetitious manner. We introduce a signal-processing based method for periodic activity detection and segmentation that utilizes a unified spatial-frequency approach.

The spectral estimation technique that we used requires a one-dimensional signal as input. In images and video, one-dimensional signals are created. We demonstrate how the more general technique of frequency estimation, object localization, and iterative decomposition using the frequency domain can be used to analyze images with periodically occurring components, video of translating images, and videos containing periodic activities.

Additionally, a method is introduced that quantifies the perceptual quality reduction in distorted images. Humans perceive distortion in images more prominently when it occurs in perceptually salient regions. This is similar to detecting periodically occurring objects, since humans will notice periodically occurring objects. Objects that occur in a periodic fashion, and whose photometric properties result in more saliency, will be more observable from a human's perspective.

We demonstrate our signal processing-based methods on railroad track inspection images, which were our primary motivation. We provide more experimental evidence of its generalization beyond this specific application.

Dedicated to my parents and my sister, for their unconditional love and support.

ACKNOWLEDGMENTS

I gratefully acknowledge everyone who has assisted, influenced, and supported me as I have created this dissertation. First and foremost, I would like to thank my adviser, Professor Narendra Ahuja, who has contributed so much to my personal and professional development. His mentorship and technical input have been invaluable, and he has constantly encouraged me to achieve more than I originally thought possible.

I thank John M. Hart and the rest of the Computer Vision and Robotics Laboratory (CVRL). The members of the CVRL have become like family to me, and I thank every one of them.

I thank Professor Christopher Barkan, Director of the Railroad Engineering Program. I would also like to thank Riley Edwards, who is instrumental in much of this work. Thank you to the Railroad Engineering Program graduate students who have worked on the Civil Engineering side of the Railroad Track Inspection project, Luis Molina and Steven Sawadisavi. It was a pleasure working with both of you. Thank you to all of the undergraduate and graduate students who have been involved in the track inspection project, and in the other projects that are a collaboration between the CVRL and the Railroad Engineering Program.

I am grateful for my doctoral committee, which consists of Professor Narendra Ahuja, Professor Christopher Barkan, Professor Thomas Huang, and Professor Douglas Jones. Thank you for all of your insight and comments.

I would also like to acknowledge all of the wonderful organizations and resources that I have been involved with here at UIUC. My graduate career is much richer due to my extracurricular involvement. I would like to acknowledge all of the people in Research Park, along with Illinois Ventures and the Technology Entrepreneur Center (TEC), for encouraging me with Fashion Latte, Inc. I would also like to thank people who have encouraged me to be involved with the campus community. Specifically, I thank the Morrill

Engineering Program (MEP), the Society of Hispanic Professional Engineers (SHPE), and the Office of Minority Student Affairs (OMSA). My graduate career has been made richer by everyone I have met along the way.

Last but not least, I am forever thankful for the encouragement and support that I have received from my parents and my sister. Throughout my life, my parents have tried their best to instill in me solid values, such as hard work and education. They have never wavered in their support, and I appreciate this. I thank my sister for being a positive role model. I would also like to thank my boyfriend, Galen, who has been my sounding board for ideas, complaints, and laughter. I could not have made it without all of the positive support I have received from everyone in my life.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

dB	Decibel
DMOS	Difference Mean Opinion Score
FFT	Fast Fourier Transform
FRA	Federal Railroad Administration
GPS	Global Positioning System
HCI	Human Computer Interaction
HVS	Human Visual System
IQM	Image Quality Measure
JND	Just Noticeable Difference
LS	Least-Squares
MSSSIM	Multi-Scale Structural Similarity
MUSIC	Multiple Signal Classification
OTR	Over-the-Rail
PSNR	Peak Signal-to-Noise Ratio
QA	Quality Assessment
RIQM	Regional Image Quality Measure
RMSE	Root Mean Square Error
SNR	Signal-to-Noise Ratio
SPIQA	Segmentation-Based Perceptual Image Quality Assessment
STFT	Short-Time Fourier Transform
VIF	Visual Information Fidelity

LIST OF SYMBOLS

β	Learned weights for SPIQA
$F(\vec{\omega})$	Fourier transform of the two-dimensional image $I(x, y)$
Γ	Sobel gradient energy
H	Histogram
$I(x, y)$	Two-dimensional image
$I_t(x, y)$	Two-dimensional video frame at time t
$\Lambda(\phi, \theta)$	Measure of periodicity of image, where the image is filtered by orientation ϕ in direction θ
κ	Weights used per-segment in SPIQA
L	Number of objects in a video
N_y	Height, in pixels, of image or frame
N_x	Width, in pixels, of image or frame
NM	Normalized mutual information measure
$\mathbf{R}$	Autocorrelation matrix
$o_l(\vec{x})$	Object, modeled in the spatial domain, for video.
$O_l(\vec{\omega})$	Object, modeled in the Fourier domain, for video.
T	
$V_{noise}(\vec{\omega})$	Noise in video, in the Fourier domain
$\vec{x}$	Two-dimensional spatial coordinates. Equivalent to (x, y).

CHAPTER 1

INTRODUCTION

Periodically occurring components are often encountered in infrastructure inspection. For example, a railroad track is composed of many individual ties, and a train is composed of individual railcars. Most repeating components are similar to each other, but not identical due to various manufacturing differences and environmental conditions. Railroads are vital to the infrastructure of most countries, but many inspection tasks are performed manually by a human inspector. Computer vision algorithms are useful in several railroad tasks, including track inspection [1], [2], [3]. Other problems involving infrastructure inspection could also benefit from computer vision.

Algorithms can potentially provide a more objective assessment of track conditions than human inspectors. However, it is difficult to create an algorithm that is robust to numerous unforeseen conditions. Spatial templates and other application-specific detection methods can be developed to accomplish specific inspection tasks. There is great value in creating a general method to detect and segment the occurrences of periodic objects in track inspection without prior knowledge of component appearance. By detecting periodic components without prior knowledge of spatial appearance, a computer vision system may one day perform track inspection over thousands of miles of track with minimal human involvement.

We first give an overview of computer vision for railroad inspection. We then describe the railroad track inspection project, and provide our motivation for studying periodic object detection in depth. We also describe our work in periodic activity recognition in video, and describe how it relates to periodic object detection.

Overall, there is a broader class of problems that this thesis presents. Spectral estimation is at the core of our algorithms, and spectral estimation techniques are applied to one-dimensional signals that are generated from images and video. This can be utilized in images with periodic components, videos with translating objects, and videos containing periodically moving objects. This thesis begins by presenting computer vision solutions for railroad inspection applications, and proceeds to develop a broader technique that can be used in the future for a variety of inspection applications.

1.1 Computer Vision for Railroad Inspection

Many railroad inspection applications can benefit from computer vision. Railroad inspection is typically performed by a human inspector. Manual inspection is time-consuming, and the Federal Railway Administration (FRA) has many guidelines that must be met for track and railcars to remain operational. Computer vision can objectively analyze large amounts of data in an efficient manner, and can supplement the manual inspection by either detecting defects, or identifying probable defects to an inspector for further inspection.

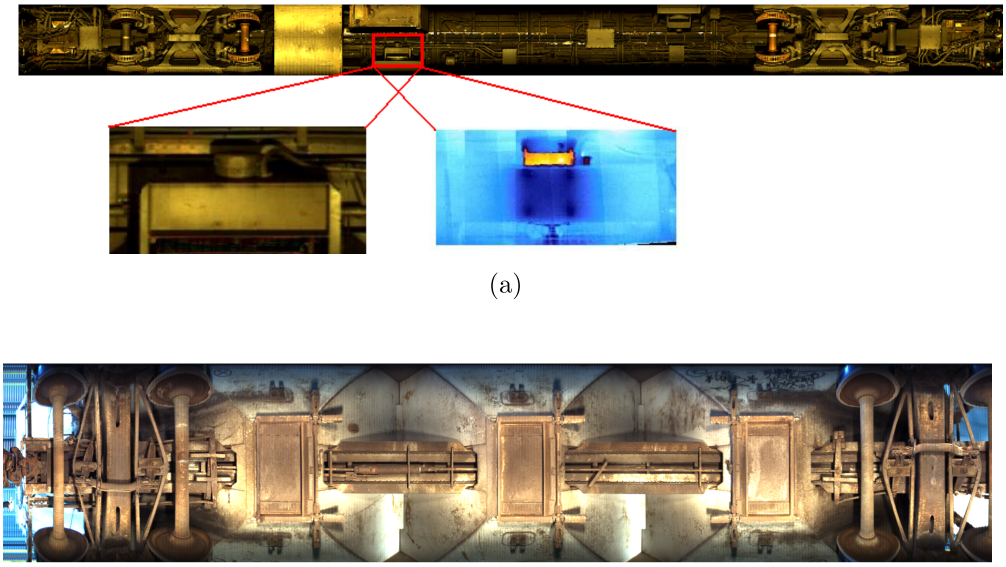
Computer vision has been successful in a feasibility study for undercarriage inspection of passenger railcars [4] (Figure 1.1(a)) and structural underframe inspection of railcars [5] (Figure 1.1(b)). For both of these applications, a panoramic image was formed from consecutive video frames, and this image was compared against specific spatial templates to ensure compliance to safety standards.

Computer vision has also been applied to an intermodal train monitoring system [6], safety appliance inspection [7], and three-dimensional track reconstruction. Computer vision has recently been applied to railroad track inspection [1], [2], and this application is the motivation for this thesis.

1.2 Computer Vision for Railroad Track Inspection

A computer vision algorithm has recently been created to identify defective track components [1] [2], [3]. The defective track components that are identified include missing and raised spikes, moved and missing anchors, and faulty turnout components.

Recently, a track cart has been developed as a collaboration between the Railroad Engineering Program and the Computer Vision and Robotics Lab-



(a)

(b)

Figure 1.1: (a) Multispectral inspection of passenger railcars and (b) structural underframe inspection.

oratory. This track cart, which is shown in Figure 1.2, captures video of a railroad track with off-the-shelf cameras, and records this data to a laptop. This allows us to record large amounts of video data that we use to detect defects in track components. Railroad track inspection algorithms are developed to analyze the data acquired from the track cart. The resulting images and video from this inspection will be presented in this thesis.

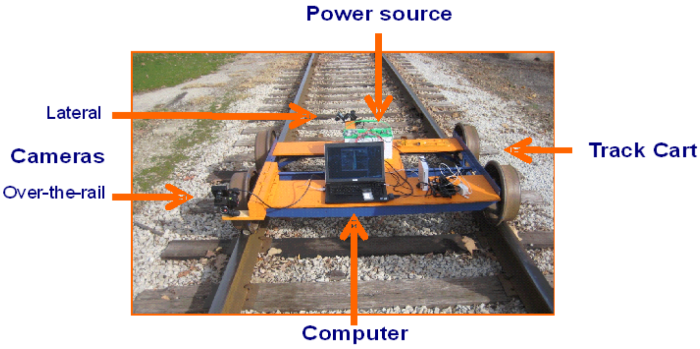


Figure 1.2: Track cart.

A necessary step to accomplish the defect detection is to have reliable detection and segmentation of track components. Though computational speed is not a requirement, it should be considered in designing an algorithm that may one day have to be adapted to perform in real time.

An additional consideration for an algorithm is its robustness to changing environmental conditions, changing track conditions, and changing appearances of track components. Since some track must be inspected for defects as often as twice per week [1], an algorithm must operate correctly for a wide range of component appearance, as track can persist for thousands of miles. Also, this data must be processed in an efficient manner. Thus, a fast, robust method that adapts to many track conditions is desired.

1.3 Periodic Object Detection

Throughout the process of algorithm development, coarse-to-fine algorithms consistently performed well. Additionally, algorithms based on textures and signal processing techniques increased the robustness of defect detection. To detect wooden ties in a track video, a method that formed global, periodic signals from the inspection data continued to be superior. These signals allowed the algorithms to achieve a more global detection, and spectral estimation was applied to the signals.

Spectral estimation is a signal processing-based method to detect periodicity in a one-dimensional signal. In this thesis, we first demonstrate the ability of spectral estimation to detect periodically repeating components that repeat in a single, horizontal direction. This is motivated by the track inspection video data that we acquired in the field, where a lateral view captures periodic ties in a unidirectional manner. We adapt this to the problem of detecting a periodic object of unknown appearance, and unknown direction of periodicity within a two-dimensional image.

The track inspection application restricts the direction of periodicity to the horizontal direction; however, many other applications do not have such specific domain knowledge. The horizontally oriented periodic object detection algorithm is expanded into a more general algorithm for detecting periodic objects in images. For this more general algorithm, we assume that periodicity can occur in any arbitrary direction, and we assume that there is no prior knowledge of the size, shape, or appearance of the repeating object.

1.4 Periodic Activity in Video

Many biological motions are periodic due to the nature of motion in biological organisms. Video obtained of such motion will produce spatial frequencies whose instantaneous frequency is periodic. Thus, signal detection can again be used to detect such objects. This is demonstrated in Chapter 5 on videos of human activity. Activity recognition from video is an important task with applications in surveillance and human computer interaction (HCI). We propose a method that models a non-rigid human with a free range of motion as a rigid object with piecewise linear motion. This allows us to detect the most dominant motions, detect common motions between activities of the same class, and detect characteristic poses for certain activities.

Although detecting periodic activity in video seems like a very different application than periodic object detection on railroad tracks, there are several similarities. Both transform the data into one-dimensional signals, and subsequently use spectral estimation to detect periodicity in those one-dimensional signals. Also, in both images and video, many of those one-dimensional signals were orthogonal to one another. Finally, both of these methods achieved superior results when they utilized techniques that were inspired by the Fourier slice theorem.

From this knowledge, we develop a more general framework. Utilizing spectral estimation is helpful when one can transform the multi-dimensional problem into a one-dimensional signal. To do this, one can use spatial projection to detect objects in images, time-elapsed spatial frequency components for detecting translating objects in video, and instantaneous frequency of time-elapsed spatial frequency components for detecting periodically repeating objects. This is covered in Chapter 6.

1.5 Perceptual Image Quality Assessment

Oftentimes, our algorithms for detecting periodicity in images will hone in on an object which a human would not notice, but which is in fact periodically repeating. Further, any human inspector is somewhat vulnerable to objects that are salient to him or her, but are not relevant to the inspection. Alternately, if algorithms are not detecting objects that are salient for a human, in certain applications the algorithms might need to be improved. We determine that it is useful to quantify how the human visual system (HVS) perceives image distortion so that in the future, our algorithm might be even more robust [8].

To quantify the loss of image quality in a way that mimics the HVS, it is intuitive to quantify the loss of visual quality to the objects in the image, since these objects will be the focus of the HVS. This would require precise object segmentation, which is cumbersome. However, mid-level region features provide a way to quantify the loss of quality for individual regions. We develop a method for quantifying the quality loss of individual regions, and then weigh the influence of these regions on the final perceptual quality metric based on each region's size and saliency. The effect of the image quality loss on structured object parts is thus effectively captured.

1.6 Overview

In this thesis, we will focus on detecting and segmenting periodically occurring objects in images. First, we focus on railroad specific applications in Chapters 2 and 3. Chapter 3 focuses on detecting periodic components that occur when repeating objects are located only along one dimension. This is expanded into a more general method for detecting periodic objects that occur along any two-dimensional orientation within an image in Chapter 4. Additionally, no prior knowledge of object size, shape, or appearance is assumed.

Periodically occurring activities in video are analyzed in Chapter 5, and a signal-processing method is used to detect and segment moving objects that are performing a periodic activity. Although the detection and segmentation of periodic objects in images, and the detection and segmentation of periodic activities in video are approached in separate methods, they in fact share many common traits. A more general method is presented in Chapter 6 which relates the detection and segmentation of periodic objects in images, translating objects in video, and periodically moving objects in video.

Computer vision inspection algorithms may one day replace or supplement manual inspection. In anticipation of this, properties of the human visual system are studied in Chapter 7. A perceptually-based image quality assessment is presented there that relies on a segmentation-based methodology.

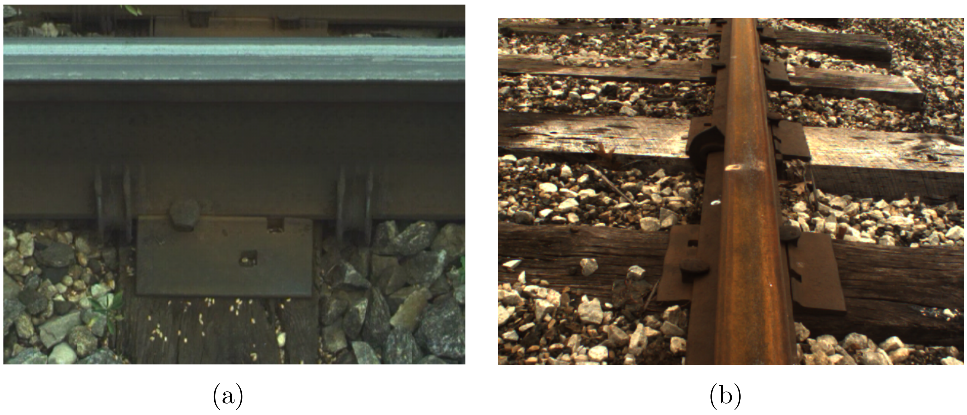
Finally, in Chapter 8, we provide suggestions for future work and conclude.

CHAPTER 2

RAILROAD TRACK INSPECTION

2.1 Track Inspection

The Federal Railroad Administration (FRA) requires track to be inspected for physical defects at specified time intervals, which may be as often as twice per week [1], [2]. Enhancements to the current manual inspection process are possible using computer vision. Computer vision could potentially supplement manual inspection due to its ability to objectively process large amounts of video and image data. Figure 2.1 shows the two camera viewpoints that we use for railroad track inspection. The viewpoint shown in Figure 2.1(a), where the side of the track is visible, is known as a lateral view. The viewpoint shown in Figure 2.1(b), where both sides of the track are visible, is known as an over-the-rail view.



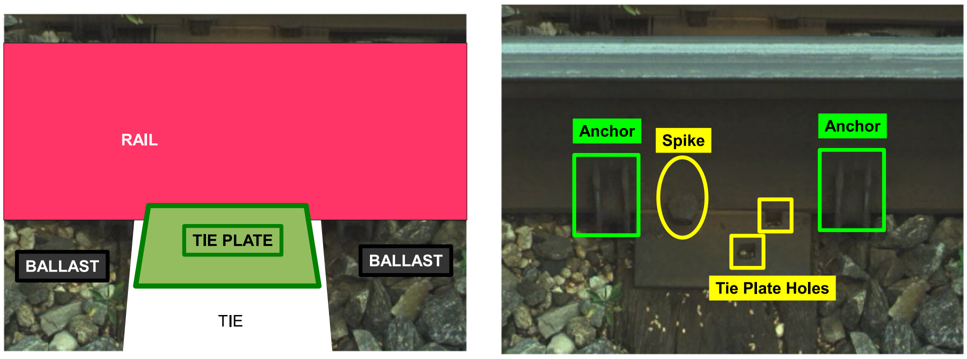
(a)

(b)

Figure 2.1: (a) Lateral view of the track and (b) over-the-rail view of the track.

2.2 Components

Figure 2.2 introduces the various components of the track. Figure 2.2 (a) delineates the largest components. The rail is in the top half of the image. The rail is the part of the track along which the train wheels move. The wooden tie is delineated with a white trapezoid, and the ties are oriented perpendicularly to the rail. The steel tie plate is delineated with a green trapezoid. Tie plates are placed between the tie and the rail when they intersect and hold the rail to the tie. The ballast, labeled on the left and right sides of the tie, is composed of small rocks.



(a)

(b)

Figure 2.2: (a) Localization of rail, ballast, tie, and tie plate. (b) Localization of spikes, tie plate holes, and anchor.

The following objects are localized in Figure 2.2 (b): one spike (shown here in an ellipse), two tie plate holes (shown in the squares), and two anchors (shown in a green rectangles). Spikes are hammered into the tie plate to keep it in place. Rail anchors secure the rail from moving perpendicular to the tie.

The components shown in Figure 2.2 (rail, tie plates, ties, cut spikes, rail anchors, and ballast) are commonly inspected for compliance with FRA regulations. Computer vision is a potentially valuable tool to facilitate in that inspection. Cumulative information on spikes, anchors, and tie plates conditions could be analyzed with pattern recognition algorithms, and defects could be detected with trend analysis.

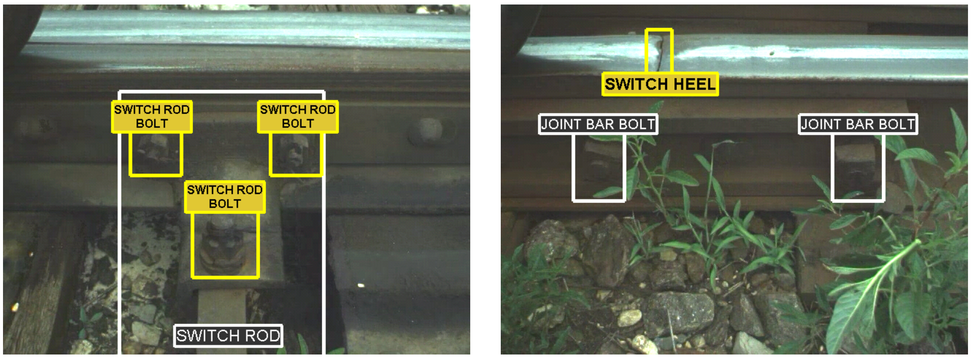
2.2.1 Turnouts

At certain locations in a track, there is convergence as tracks join each other and divergence as one track forks into two. These areas of the track are known as turnouts or switch areas, and defects in the switch area will frequently result in an accident. The part of the track that occurs before and after the switch area is known as the closure area. As a railcar enters the closure area, the switch must be in a configuration that allows the railcar to continue to the correct track. There are many critical components within the switch area.

Track components in turnouts differ in both size and shape from those found in normal tangent or curved track. For this reason we must correctly identify the specific section of the track the system is inspecting and whether it is part of a turnout. Some of the components located within the turnout are: the switch point, the frog, the heel, and the joint bar. Worn or broken switch points are the most frequent causes of mainline accidents in turnouts for track Classes 4 and 5 on U.S. railroads, followed by other frog, switch, and track appliance defects [9].

Defect detection for the switch point and other components is extremely important. However, to detect defects in the individual parts, a more global detection of the switch area itself must occur. Our goal is to detect the presence of a switch area using the lateral view. In the future, a detection using the over-the-rail view could also be achieved. Also, alternatively a global positioning system (GPS) could be implemented to detect the occurrence of a turnout. However, visual inspection would still be necessary to analyze individual components within the turnout, and a globally motivated initial step would create the most robust algorithm.

Figures 2.3(a) and (b) show some of the components for which computer vision will assist in turnout inspection. These are: the switch rod and switch rod bolts (Figure 2.3(a)), and the switch heel and joint bar bolts (Figure 2.3(b)). Computer vision should be able to achieve the necessary resolution to detect defects in these objects. Additionally, the periodicity of the bolts will aid in turnout detection.



(a) (b)

Figure 2.3: Turnout components: switch rod and switch rod bolts (a), and switch heel and joint bar bolts (b).

2.3 Panorama Generation

Panoramic images aid in visualizing defects and can be used in the future to provide a chronological record of track conditions. Algorithms generate panoramas from video data by concatenating consecutive video frames from video that is acquired as the track cart rolls along the track. The procedure is shown in Figure 2.4. When the video records motion in the horizontal direction, the vertical strips at the center of the frames provide the minimum amounts of distortion and perspective difference. The distortion becomes more severe with increased distance from the component and the center of the image. In Figure 2.4, the third step performed by the algorithm is velocity estimation, which detects the distance the camera moved between consecutive frames. This velocity information is used to determine the size of the strip required from each frame to construct accurate panoramas at a variety of data collection speeds. These strips are then appended to each other to create the final panoramic image.

Phase correlation is used to estimate the velocity in a frame-by-frame manner. Phase correlation is a well-known technique in image processing that estimates the displacement between two images by utilizing the Fourier domain. For two consecutive images, I_1 and I_2, let $I_2(\vec{x}) = I_1(\vec{x} - \vec{x}_0)$; that is, they are related by a spatial shift. Then their Fourier transforms, $F_1(\vec{\omega})$ and $F_2(\vec{\omega})$, are related by a phase shift, $F_1(\vec{\omega}) = F_2(\vec{\omega})e^{-j\vec{\omega}^T\vec{x}_0}$. For robust detection of spatial shift, one can use phase correlation of the frequency domain

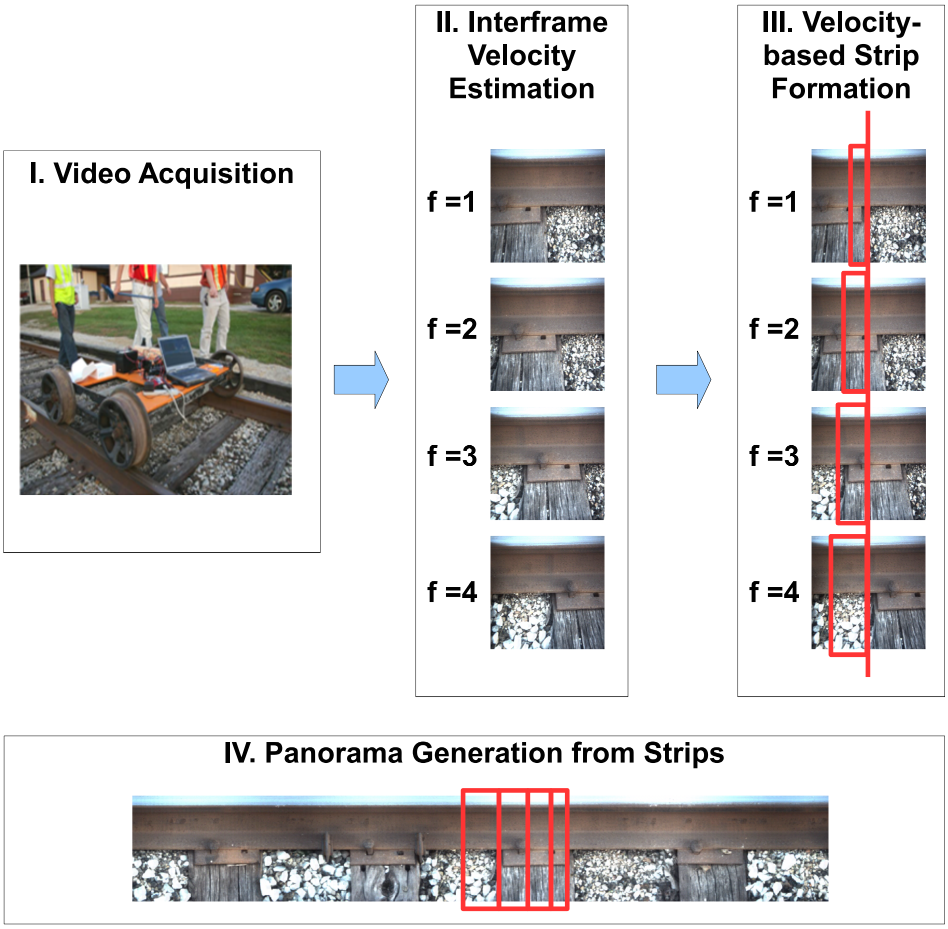

I. Video Acquisition
II. Interframe Velocity Estimation
III. Velocity-based Strip Formation
f =1
f =2
f =3
f =4
f =1
f =2
f =3
f =4
IV. Panorama Generation from Strips

Figure 2.4: Method for creating panorama.

as follows:

$$\frac{F_1(\vec{\omega})F_2(\vec{\omega})^*}{|F_1(\vec{\omega})F_2(\vec{\omega})^*|} = e^{j\vec{\omega}^T\vec{x}_0} \tag{2.1}$$

where the exact value of $\vec{x}_0$ can now be determined by transforming $e^{j\vec{\omega}^T\vec{x}_0}$ into the spatial domain, where $\vec{x}_0$ will appear as a peak $\delta(\vec{x} - \vec{x}_0)$. Each individual vertical strip's width is determined using this method, and strips are concatenated to form the panorama.

Once the panoramas are generated, the results of the component inspection can be superimposed onto it (Figure 2.12 on page 19). Alternately, the inspection can take place on the panorama itself by detecting the appropriate search areas, and subsequently recognizing the components and detecting defects.

2.4 Previous Work

We have developed algorithms to detect the rail, wooden ties, ballast, tie plates, cut spikes, and rail anchors using a global-to-local algorithmic approach [1], [2]. The components are detected using manually created models and highly customized techniques. Our approach uses low-level features such as image gradients and textures to provide robust detection of more consistent features, such as the rail, then uses these features to resolve a restricted search area to find components with greater visual variation, such as cut spikes and anchors. The spikes and anchors are then found in these restricted search areas using spatial template matching.

The local features that we use include edges and Gabor features. Edges are frequently used to detect objects in computer vision since object boundaries often generate sharp changes in brightness [10]. Image gradients (edges) should be consistent among differing ties and rails, but unanticipated track objects or debris could create unwanted edges, causing challenges for the algorithms. For this reason, texture information from the ballast, tie, and steel was incorporated into an edge-based algorithm to improve its robustness. This approach relies on texture classification using Gabor filters, which produce low-level texture features.

2.4.1 Lateral Analysis

Our method for analyzing track components from a lateral view operates in a coarse-to-fine manner where the components are detected and localized, in the following order:

1. Detect and segment the rail

2. Coarsely detect ties

3. For each tie, segment the tie plate

4. For each tie, segment the tie

5. For each tie, detect, segment, and measure the spikes and anchors

We demonstrate our method for the tie shown in Figure 2.5, which was acquired from field video data. First, we detect and segment the rail.

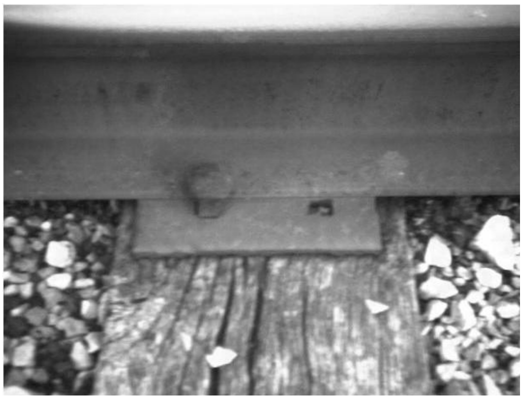
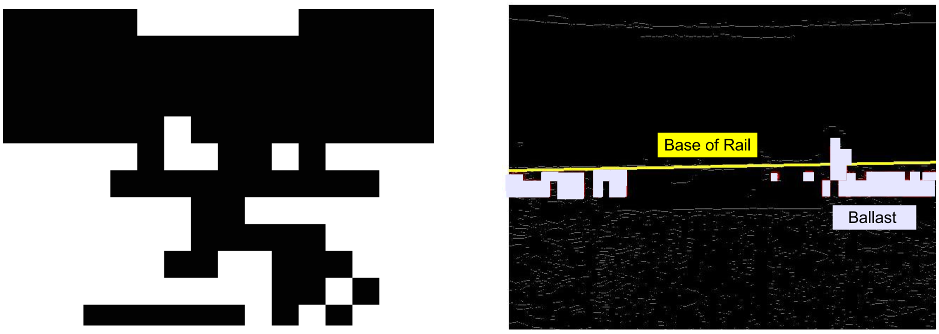


Figure 2.5: Field-acquired video frame of tie.

Since we operate using a coarse-to-fine approach, we decompose the image beginning with the rail, which is the largest, most consistently detectable object. The strong gradients of the rail make it the most detectable object in both of the camera views. The base of the rail from Figure 2.5 is localized by detecting its strong horizontal gradient. However, to prevent false detection, particularly due to any edges caused by shadows, it is necessary to use texture classification in detecting the base of the rail.

We reliably classify textures using Gabor filtering. Specifically, we use the Gabor filters described in [11]. Labeled examples of ballast, tie, and steel textures were created using previously stored images. When presented with a previously unseen image, such as the one in Figure 2.5, texture patches are extracted and classified as either "ballast" or "non-ballast" (steel or tie). Though the classification may contain noise due to occluding objects (e.g. leaves or ballast on ties), this method robustly provides a "non-ballast" region that is centered on the tie. This is demonstrated in Figure 2.6(a), where the frame is divided into block-based patches, and each patch is classified as either "ballast" (shown in white) or "non-ballast" (shown in black). This is a coarse estimate, and later stages localize the track components in greater detail.

Using this method, the rail and tie are both isolated from the ballast (Figure 2.6 (a)). Though the boundaries are inexact, the tie area is reliably isolated for subsequent processing. We continue to process a video frame if a tie is detected in the center of the frame.



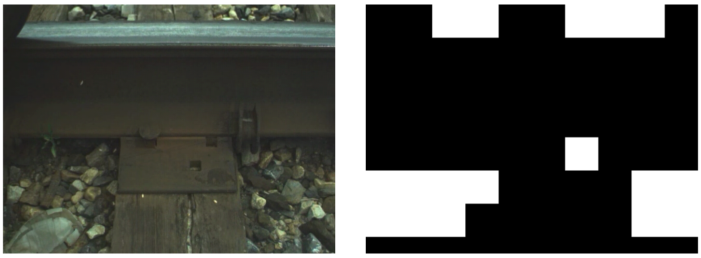
(a) (b)

Figure 2.6: (a) Ballast detection mask. (b) Localization of base of the rail.

The base of the rail is detected by a strong horizontal edge. In Figure 2.6(b), texture information is used to ensure that the detected base of the rail separates steel from ballast on either side of the tie.

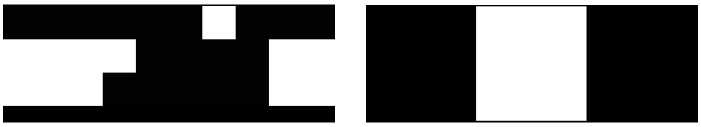
We also created an algorithm that uses global information to detect which frames in a video contain ties. For example, to determine that the image in Figure 2.7(a) contains a tie and therefore requires further processing, information about the consecutively surrounding video frames is included in that detection. To achieve this, for every video frame we quantify the probability that a tie is present. We do this by comparing each binary mask, as in Figure 2.7(b) to a binary spatial template. We only compare the lower half of the mask, as shown in Figure 2.8(a), with the template shown in Figure 2.8(b). We subtract the difference, and sum the absolute value of that difference. A value is computed for each frame in the video. This results in a signal that is sinusoidal with respect to time when the inspection video is acquired at a constant speed, as shown in Figure 2.9(a). To avoid the requirement of constant speed, the signal can also be recorded as a function of inter-frame displacement (Figure 2.9(b)).

The video frames that contain a tie are located at the maximum amplitude of the signal in Figure 2.9(a). In the case of non-constant velocity, the video frames which contain a tie are computed by determining which frame



(a) (b)

Figure 2.7: Masked lateral image (a) original image and (b) binary texture classification mask for image, where black is “non-ballast” and white is “ballast.”



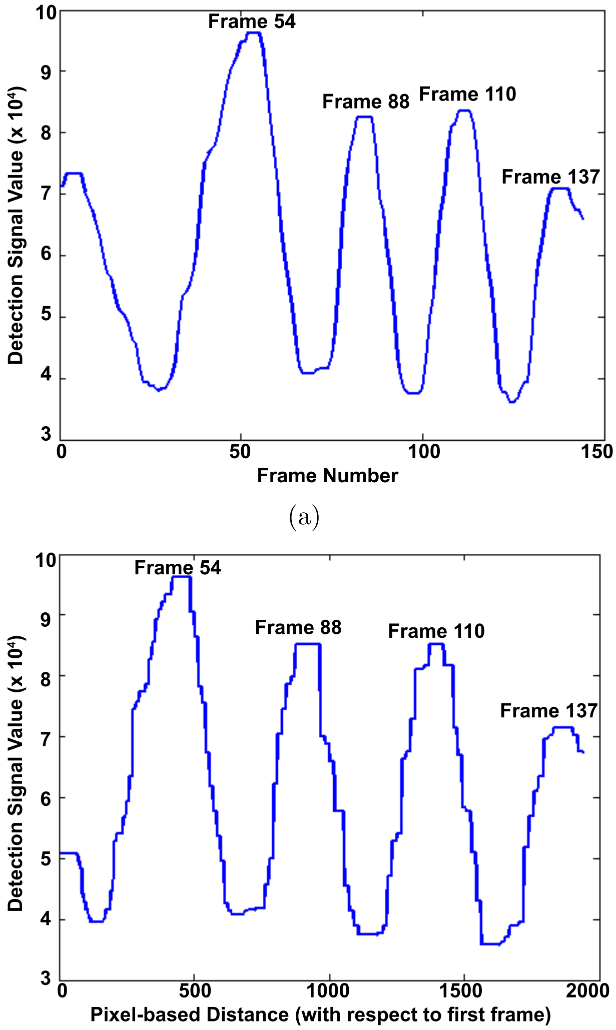
(a) (b)

Figure 2.8: (a) Lower half of binary texture classification mask and (b)template that mask is compared against.

corresponds to the pixel-based distance at the maximum amplitude of Figure 2.9(b). The remainder of the algorithm will proceed only for the frames that contain ties.

The tie plate is subsequently detected by its two horizontal edges, and texture information is again used to confirm that the upper edge separates steel-to-steel textures and that the lower edge separates steel-to-tie textures. The results are shown in Figure 2.10(a). After delineation of the two horizontal edges, the vertical edges that form the sides of the tie and tie plate are found since they are reliably detected only if their search space is restricted. A restricted search space is needed because shadows, occlusions, and other unforeseen anomalies will cause unanticipated edges and shapes. The vertical tie edge is the dominant gradient that exists on both sides of the tie plate-to-tie edge, while the vertical tie plate edge is the dominant gradient that exists only above the tie plate-to-tie edge.

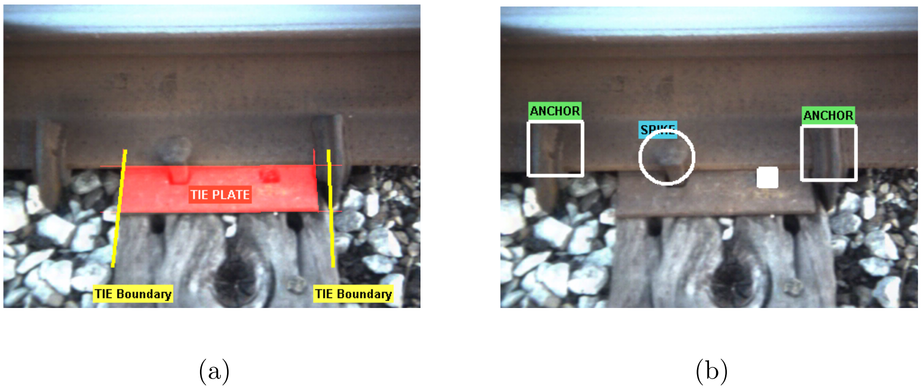
The spikes are located with spatial correlation using a previously developed template (Figure 2.10(b)). Spikes will only be found in certain locations, which will limit the search area after the tie plate and rail are both detected.



(a)

(b)

Figure 2.9: Response of binary classification mask to template as a function of (a) frame number and (b) pixel-based distance.



(a)

(b)

Figure 2.10: (a) Tie plate and tie detection. (b) Spike, anchor, and tie plate hole detection.

These locations include a row of line spikes next to the base of the rail and another row of hold down spikes further from the rail. Since the search space is restricted, a low threshold can be set for the template response. Therefore, the appearance of a spike altered by conditions has a higher probability of being detected, since we have lowered the threshold for a template match. Missing spikes are detected by a two-dimensional filter that consists of a dark square surrounded by a steel-colored square. The color of the steel is extracted from the isolated tie plate. Our current detection of spike heads is not yet robust due to environmental variability and differential wear patterns, but when the search area is limited, the accuracy improves.

Gauge side refers to the part of the track that is between the two rails, and field side refers to the part of the track outside the two rails. Rail anchors, when installed correctly, have more distinctive visual characteristics when viewed from the gauge side of the rail as compared to the field side: therefore, our anchor inspection primarily uses a gauge-side lateral view. The search area for the anchors is restricted to where the rail meets the ballast on either side of the tie plate. Anchors are detected by identifying their parallel edges, and the distances to both the tie and tie plate are measured. This scheme is robust to shadows, since shadows will result in similar intensity distributions for parallel edges in the same anchor. It is also robust to anchor rotation and skewing, since the parallel edges that we detect need not be vertical. In Figure 2.11, a defective (moved) anchor is detected due to the anchor's distance from the tie.

Panoramic images (Figure 2.12) provide a way to visualize the results of computer vision algorithms, even if the algorithms are applied to the video itself. Panoramas could be used as a recording medium to record the history of a track for trend analysis and alert human inspectors of problems.

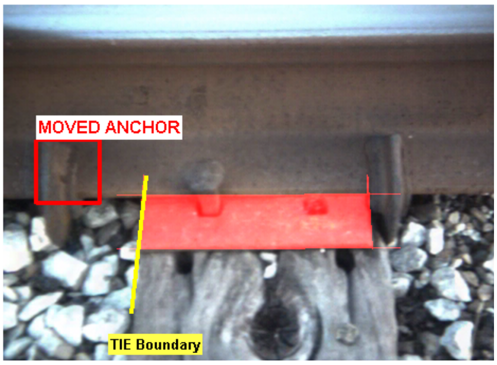
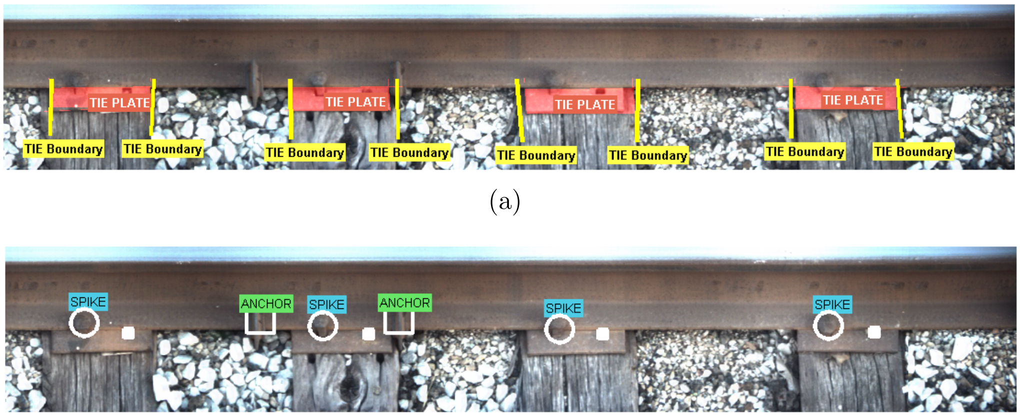


Figure 2.11: Detected defect (moved anchor).



(a)

(b)

Figure 2.12: (a) Detected ties/tie plates and (b) detected spikes and anchors on panoramic images.

2.4.2 Over-the-Rail Analysis

The algorithmic approach to over-the-rail analysis is similar to that of the lateral view. The ties are located again with a template mask. We perform our analysis on video frames that contain a foreground tie located approximately at two-thirds the height of the image. A binary image that contains binary blocks which indicate ballast/non-ballast classification is used, as shown in Figure 2.13(a). The tie filter shown in Figure 2.13(b) is overlaid against the binary image, and the amount of overlap is recorded. This filter response is plotted against the number of frames in Figure 2.14. Notice that the filter response is periodic. In the video, the foreground ties come into view in a periodic manner, and as such the filter response to the template in Figure 2.13(b) is periodic.

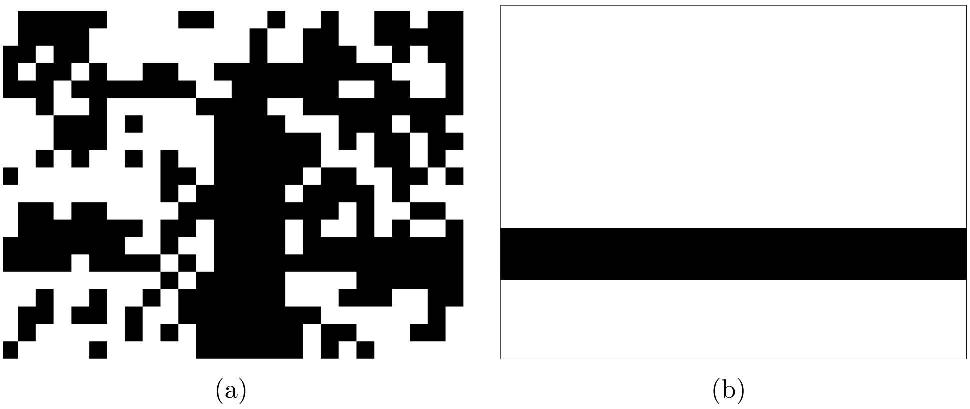


Figure 2.13: (a) Mask of ballast / non-ballast textures. (b) Tie filter.

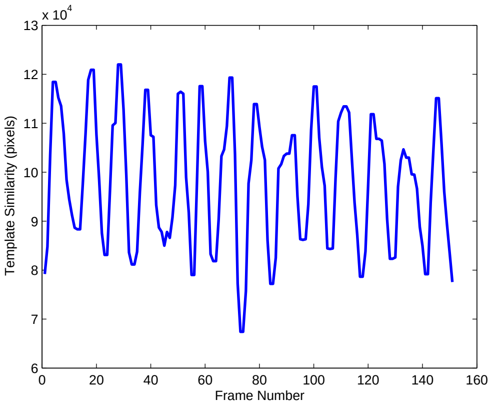
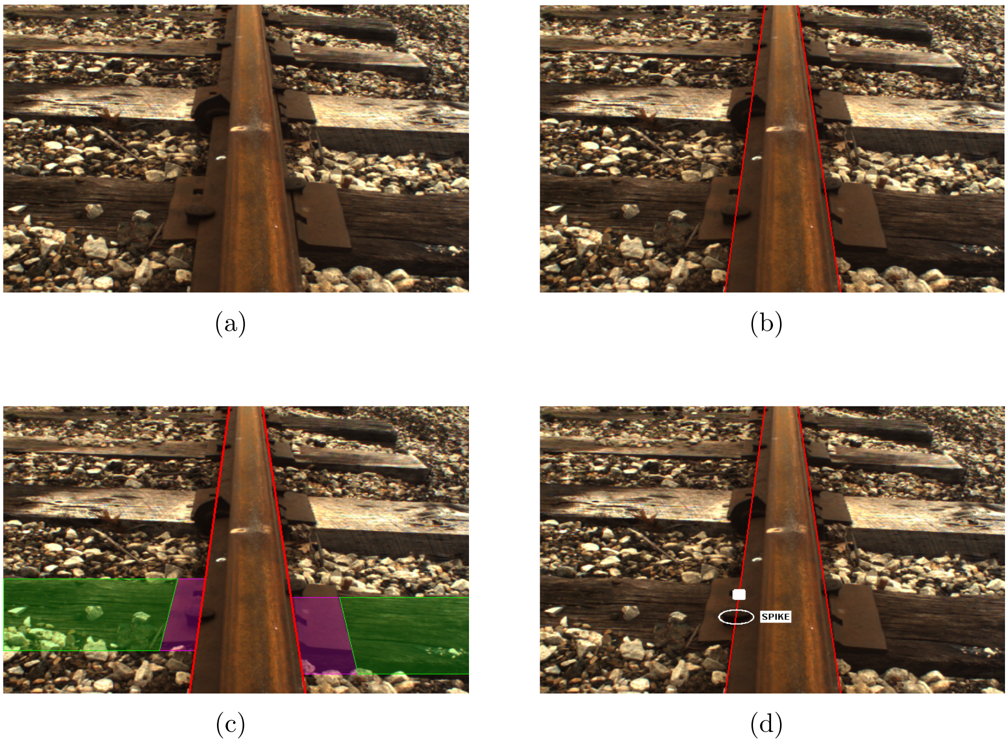


Figure 2.14: Response of mask from Figure 2.13(a) to the tie filter in Figure 2.13(b).

The rail is again considered the most reliable object in the video, and a coarse-to-fine approach is adopted which begins with the localization of the


(a)

(b)

(c)

(d)

Figure 2.15: Over the rail detection. (a) Original image, (b) detected rail, (c) tie plate and tie delineated, and (d) spike and tie plate hole detection.

rail, as shown in Figure 2.15(b). Then, the tie and tie plate are delineated. Strong gradients are hypothesized to be the edges of the tie, and the area above and below the hypothesized edge is tested to confirm that it is a boundary between tie and ballast. The resulting delineation is shown in Figure 2.15(c). The delineation between the tie plate and tie/ballast is done in a similar manner. Finally, the spikes and anchors are located with spatial and spatial gradient templates. The result is shown in Figure 2.15(d).

2.5 Signal Processing Insights

In developing inspection algorithms for the specific application of railroad track inspection, important insights were gained into how valuable signal processing techniques are in developing robust computer vision algorithms. As shown in Section 2.3, velocity of moving track was more accurately estimated and ties were more accurately detected and segmented. The velocity estimation utilized the fact that inter-frame translation produces a frequency-domain phase shift. This was used to create a panoramic image that was free of processing artifacts.

Our early work on track inspection relied on highly calibrated templates and customized algorithms [1]. One of the most challenging parts of track inspection, however, is the required robustness in the presence of varying environmental conditions and varying component variations that are found within thousands of miles of track. Thus, algorithms should be able to tolerate a wide variance of component appearances due to environmental conditions and manufacturing differences. From Sections 2.4.1 and 2.4.2, we discovered that the most robust part of this algorithm was due to creating a one-dimensional signal from the video data. We also discovered that Gabor filters were reliable in texture classification. In fact, in many of the training algorithms for texture classification, a synthetic picture of ballast was used, and the results were still acceptable. Therefore, by combining both Gabor filtering and one-dimensional signal extraction, a robust method for periodic component extraction could be created.

2.6 Conclusion

We demonstrated the effectiveness of our defect detection algorithm for railroad track inspection. We were able to successfully isolate ties, delineate ties and tie plates, and localize and measure the spikes and anchors. As noted in Section 2.5, robust algorithms can be developed that utilize signal processing-based methods. Specifically, Gabor filters have demonstrated robustness in characterizing textures, and the algorithm that will be presented in Chapters 3 and 4 utilizes Gabor filters to decompose images so that the repeating objects are more detectable. Also, as noted in Section 2.5, a one-dimensional signal mapping produces a more global scope for detection, and in Chapters 3 and 4, we will demonstrate how spectral estimation can be applied to one-dimensional signals.

CHAPTER 3

DETECTION AND SEGMENTATION OF PERIODICALLY REPEATING OBJECTS WITH KNOWN ORIENTATION

In railroad inspection, periodically occurring components are often encountered. For example, a train is composed of individual railcars, and railroad track is composed of many individual ties. Most repeating components are similar to each other, but not identical due to various manufacturing differences and environmental conditions. Railroads are vital to the infrastructure of most countries, but many inspection tasks are performed manually by a human inspector. Computer vision algorithms have shown promise in solving many railroad problems, including track inspection [1].

An algorithm is presented in this chapter for detecting and segmenting periodically occurring components. We use spectral estimation algorithms on a Gabor feature space to identify repeating textures and components. This technique is demonstrated on railroad track inspection panoramas. An example panorama is shown in Figure 3.1, where each wooden tie occurs at four approximately equidistant locations. In addition to finding the dominant periodicities and components, spectral estimation algorithms allow us to detect less-dominant periodicities. This is useful since many components are themselves made of smaller periodically repeating components.

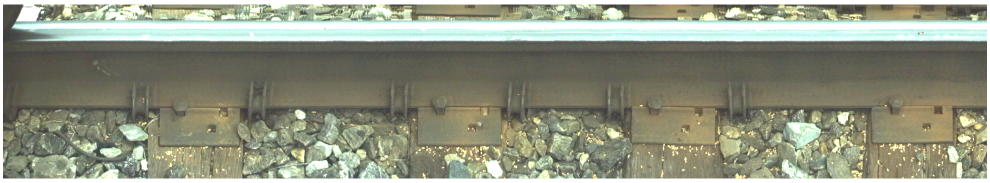


Figure 3.1: Rail track inspection panorama.

3.1 Background

The Fourier domain is effective in classifying spatially periodic textures [12]. The spatial frequency can be used as a self-filter, where the highest magnitude spatial frequencies from the image itself are used in filtering, thus emphasizing the parts of the spatial frequency that contain a dominant periodicity. However, this method is ineffective on many real world images, such as a railroad track panorama. There are two main reasons for this: first, the tie and ballast textures are not well-structured. Thus, quantifying them as periodically repeating tessellations is inaccurate. Second, the illumination in both the tie and ballast have similar magnitudes, and only larger gradients could benefit from self-filtering (e.g. rail-to-ballast).

Periodically occurring objects are not often studied in literature, but many existing object detection methods could be applied to periodic objects. Typically, object detection is performed by training a system to detect the appearance of a certain object, and then testing the system on a database of previously unseen objects. Existing methods often use spatial feature point detection followed by hypothesis testing of proposed models [13]. Often, one needs to find both the spatial appearance and configuration description of object parts that are shared by objects of the same class. This could be extended to periodically occurring components by using feature detection followed by hypothesis testing of various hypothesized periodicities. The novelty of our algorithm lies in its ability to find the component descriptions that best describe the periodically recurring parts of an image through spectral estimation techniques, which provide elegant solutions for detecting multiple periodic signals in a noisy signal. In this chapter, the images will contain periodically occurring objects in a previously known direction. By applying this to a Gabor feature space, we are able to detect periodically occurring textures and components in an unsupervised manner.

3.2 Model

Periodically occurring track components are difficult to detect due to their within-class variation, the hierarchical structure of many components, and aberrations to exact periodicity. A successful methodology should (1) estimate the periodicity for each component, (2) describe the spatial appearance of those components, and (3) spatially localize those components.

Our approach begins with finding dominant spatial periodicities using spectral estimation techniques. We then determine a spatial description for the components using that periodicity.

For simplicity, we have applied this only to images with repeating components in the horizontal direction, but it could be applied along other orientations as well. Each image $I(\vec{x})$ is decomposed into R groups of rows, so that $I(\vec{x}) = [I_1(x)^T I_2(x)^T \ldots I_R(x)^T]^T$. Each group of rows, $I_r(x)$, contains $\frac{N_y}{R}$ pixel-high image rows and is represented as the summation of K repeating components C_k where $k \in K$, each occurring with periodicity T_k. Each component is described by $C_k = \{\mathbf{p}_k, T_k, \upsilon_k\}$, where $\mathbf{p}_k$ is the two-dimensional appearance, T_k is the periodicity, and υ_k is the phase offset. Periodicity is defined as the number of pixels between consecutive component occurrences, as measured from a fixed point in that component. Phase offset indicates the location of the first instance of a repeating component with respect to the left-most pixel at $x = 0$.

We express the image row $I_r(x)$ with the equation

$$I_r(x) = \sum_{k=1}^{K} \sum_{z \in Z_k} \delta(x - (x_k + zT_k + \upsilon_k))\mathbf{p}_k(x_k) + \eta(x)) \tag{3.1}$$

where Z_k represents the range of z for which the component instance is contained within the image row $I_r(x)$, and $\eta(x)$ is the part of the image row that does not repeat. The term x_k refers to each component's internal coordinate system, which indexes into $\mathbf{p}_k$ using only one dimension. We assume that the image itself is composed primarily of repeating components $\mathbf{C}$.

Our proposed method utilizes spectral estimation techniques to detect dominant periodicities, and then subsequently finds the most probable υ_k given the dominant T_k. We assume that the appearance description $\mathbf{p}_k$ has width $(\frac{T_k}{2})$, so $\mathbf{p}_k$ can be inferred from T_k, υ_k, and the image $I(\vec{x})$.

The entire algorithm is shown in Figure 3.2. As a first step (top of Figure 3.2), the image row $I_r(x)$ is converted into a sequence of M blocks, $[b_0(x)b_1(x) \ldots b_M(x)]$. Gabor features are extracted for each block, which we represent as $\mathcal{G}_d(b_m(x))$, where d is the dimension of the Gabor feature. The

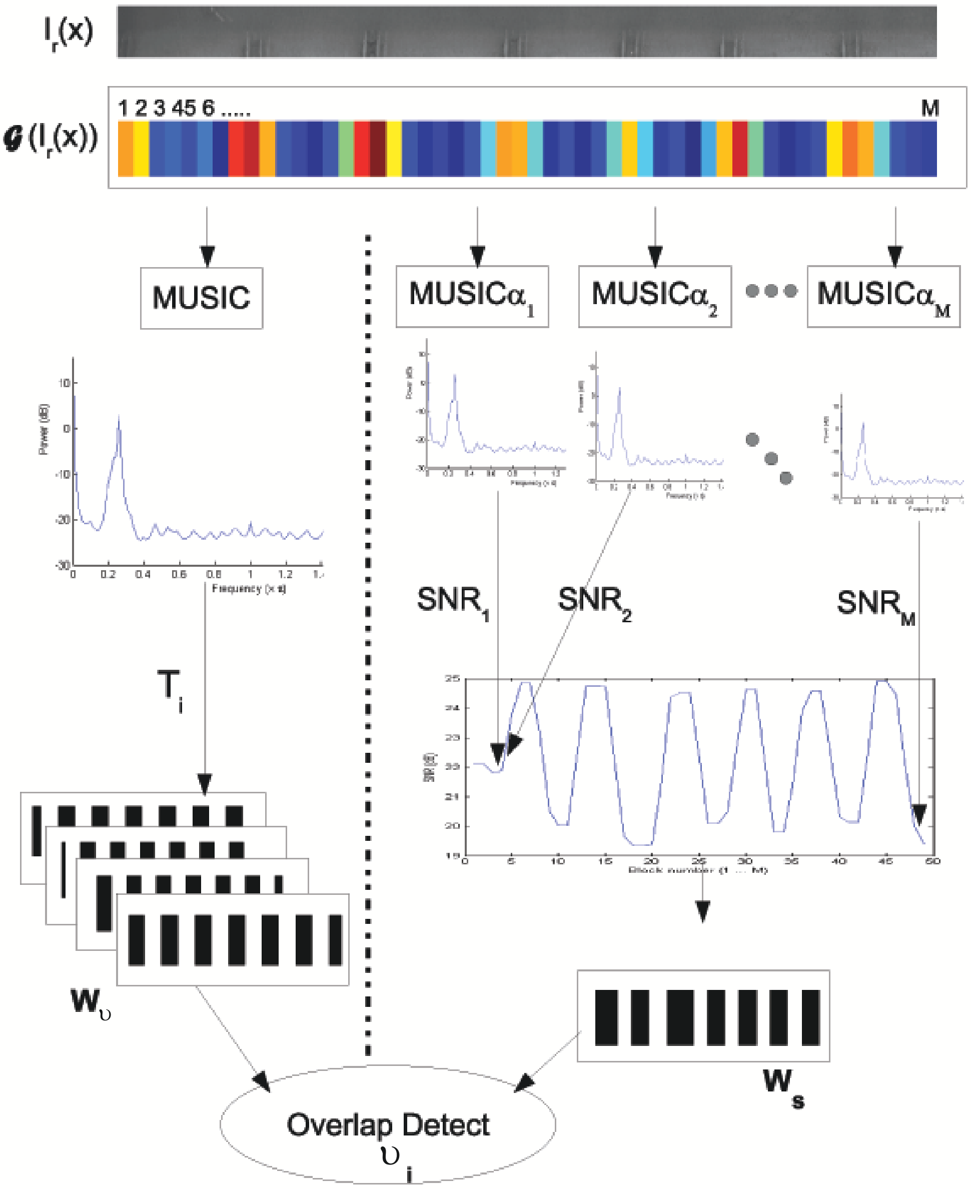


Figure 3.2: Component localization algorithm.

Gabor features from [11] were used, since they are effective at discriminating textures. The cumulative application of the d-th dimension of the Gabor transform to all blocks in the image row $I_r(x)$ is represented as $\mathcal{G}_d(I_r(x))$. The total number of dimensions is G. A G-dimensional Gabor feature space is utilized for its descriptive capabilities [10]. Illumination alone is prohibitively noisy due to the high variance of color and intensity, even within several consecutive instances of the same component.

The remainder of the algorithm will be described in Sections 3.3 and 3.4. In Section 3.3, the periodicity of each row is estimated. In Section 3.4, the components are localized by computing the phase offset υ_k for all components, and subsequently $\mathbf{p}_k$ is computed.

3.3 Detection Using Multiple Signal Classification

The row-wise frequency estimation is accomplished by applying the Multiple Signal Classification (MUSIC) algorithm to the block-wise Gabor transformed image rows of $\vec{b}(x)$. The MUSIC algorithm is used to detect multiple signals contained in a received signal [14]. The algorithm models a received signal, $\mathbf{y}$, as

$$\mathbf{y} = \mathbf{A}\mathbf{s} + \mathbf{v} \tag{3.2}$$

where $\mathbf{A}$ is the signal subspace, $\mathbf{s}$ is the vector of signal amplitudes with respect to that subspace, and $\mathbf{v}$ is a noise vector. In MUSIC, the dominant signals, $\mathbf{s}$, are computed by finding the dominant frequencies on the noise projection subspace. First, the covariance matrix $\mathbf{R}_\mathbf{y}$ is computed

$$\mathbf{R}_\mathbf{y} = E\{\mathbf{y}\mathbf{y}^H\} = \mathbf{Y}\mathbf{Y}^H = \mathbf{A}\mathbf{R}_\mathbf{s}\mathbf{A}^H + \sigma^2\mathbf{I} \tag{3.3}$$

where $\mathbf{Y}$ is a rectangular Toeplitz matrix such that $\mathbf{Y}\mathbf{Y}^H$ is a biased estimate of the autocorrelation matrix for the signal vector $\mathbf{y}$ [15]. The matrix $\mathbf{Y}$ is defined as

$$\mathbf{Y} = \begin{bmatrix} y(h+1) & \cdots & y(1) \\ \vdots & \ddots & \vdots \\ y(M+h) & \cdots & y(h+1) \\ \vdots & \ddots & \vdots \\ y(M) & \cdots & y(M-h) \\ y^*(1) & \cdots & y^*(h+1) \\ \vdots & \iddots & \vdots \\ y^*(h+1) & \cdots & y^*(M-h) \\ \vdots & \iddots & \vdots \\ y^*(M-h) & \cdots & y^*(M) \end{bmatrix} \tag{3.4}$$

where M is the length of the block-wise signal $\mathbf{y}$, and h is an index $1 \leq h \leq M$.

We create independent signals from each of the signals $\mathbf{y}_d = \mathcal{G}_d(I_r(x))$. To estimate periodicity for the entire row, we can either predict the periodicity for each signal, $\mathbf{y}_d = \mathcal{G}_d(I_r(x))$, and choose the period which is computed the most often, or we can predict the periodicity using all of the inputs, $\mathbf{y}_d$ combined. We found that using the latter approach yielded the more robust estimations.

We sum the $\mathbf{Y}$ matrices from each of the dimensions, where a $\mathbf{Y}$ matrix composed of the response from the d-th dimension is denoted $\mathbf{Y}_d$. The final matrix $\mathbf{Y}$ is the computed as

$$\mathbf{Y} = \sum_d \mathbf{Y}_d \tag{3.5}$$

We initially estimate the number of signals (or components) that are present in the current row as L, and the number of eigenvectors in the decomposition of $\mathbf{R}_y$ as D. The eigenvectors $\mathbf{e}_j$, where $j \in \{(L+1)\ldots D\}$, span the noise subspace. This is used to find the maximum values of the following function:

$$J(\omega) = 20\log_{10}\left(\frac{1}{\sum_{j=L+1}^{D} |\mathbf{a}^H(\omega)\mathbf{e}_j|^2}\right) \tag{3.6}$$

which allows us to implement $J(\vec{\omega})$ as the sum of the fast Fourier transform

(FFT) of all noise eigenvectors e_i where $i \in \{(L+1)\dots D\}$. From this, we can detect up to L periodicities that are present along $I_r(x)$, as they correspond to L peaks. We then detect the peaks of $J(\omega)$ based on SNR. An example output of $J(\omega)$ is shown at the output of every MUSIC block in Figure 3.2, where power (in dB) is plotted as a function of frequency ω. Peaks are measured using SNR with respect to the noise floor. We define the SNR of each peak detect for T_k as

$$SNR_k = J\left(\frac{2\pi}{T_k}\right) - \sum_{\omega \neq \frac{2\pi}{T_k}} J(\omega) \tag{3.7}$$

where $\frac{2\pi}{T_k}$ is quantized to the nearest value of ω. In our experiments, ω is defined in increments of $\frac{2\pi}{256}$.

3.4 Component Localization

The output of Section 3.3 provides an estimate for periodicity, T_k, but no information on the location of a component. We define υ_k as the phase offset from the left-most coordinate $x = 0$ (the left-most position of $I_r(x)$). To find T_k, we create a series of masks, each with periodicity T_k, which consist of alternating ones and zeros with an offset by a candidate υ. Each mask, $\mathbf{W}_\upsilon$, can be placed over the panorama, where each $\mathbf{W}_\upsilon$ is composed of vertically aligned columns of ones and zeros that alternate every $(\frac{T_k}{2})$, with an offset of υ pixels from $x = 0$. Our goal is to find υ_k that isolates the periodically occurring component $\mathbf{p}_k$. This is shown at the bottom of Figure 3.2.

We propose a weighted MUSIC algorithm, MUSIC$_{\alpha_m}$, to test the hypothesis that any given b_m belongs to a periodically repeating component. As shown in Figure 3.2, the results of MUSIC$_{\alpha_m}$ are combined with estimates of $\mathbf{W}_\upsilon$ to determine υ_k. We do not produce a single $\mathbf{p}_k$ for each C_k, but rather we produce a set of all appearances of that component. One could pick one representative $\mathbf{p}_k$, or use further modeling techniques to fuse them into a single appearance model.

3.4.1 Weighted MUSIC Algorithm

We assume that all occurrences of the same component are clustered in Gabor feature space, so that if a block b_m periodically occurs, then the distance in feature space between $b_m(x)$ and all $I_r(x) = [b_0(x)b_1(x)\dots b_M(x)]$ will oscillate when computed in successive $0, 1, \dots, M$ order. We use the value of block $b_m(x)$ in Gabor feature space to weight the G-dimensional feature space by the vector α, where $\alpha_{m,d} = ||\mathcal{G}_d(b_m)||$, where $\mathcal{G}_d(b_m)$ refers to the d-th dimensional Gabor features of b_m.

Equation 3.5 is reweighted for each Gabor block m so that

$$\mathbf{Y} = \sum_d \alpha_{m,d}\mathbf{Y}_d \tag{3.8}$$

The resulting SNR_m is plotted for M components, as shown in the right side of Figure 3.2. Blocks that periodically occur with period T_k should produce the highest SNR.

3.4.2 Component Isolation

After MUSIC$_{\alpha_m}$ is computed $\forall m \in M$, the blocks $b_m(x)$ that produced highest SNR$_m$ are hypothesized to belong to the periodically occurring components. A new window is formed from this detection, $\mathbf{W}_s(x)$. The window mask $\mathbf{W}_\upsilon(x)$ with maximum overlap to $\mathbf{W}_s(x)$ is chosen as the optimum υ_k.

The spatial description $\mathbf{p}_k$ is the description of a repeating component that is found by extracting all occurrences of a periodically repeating component. Once T_k and υ_k are found, then either a representative component is chosen or the median value of all components is computed.

3.4.3 Non-Dominant Periodicity Detection

One of the advantages of the spectral estimation framework is its ability to detect multiple periodicities. In the previous methodology, the dominant frequency will in fact consist of the first two peaks, since the input $\mathbf{y}$ is real. The previous methodology can be performed on non-dominant frequencies by setting $L > 2$ and detecting a third peak in $J(\omega)$. In our algorithm, we limit the number of detected peaks in the first iteration to $L = 1$. After isolating the component with primary periodicity, the algorithm is rerun to detect the less dominant of $L = 3$ periodicities. Section 3.5 contains an example with multiple periodicities.

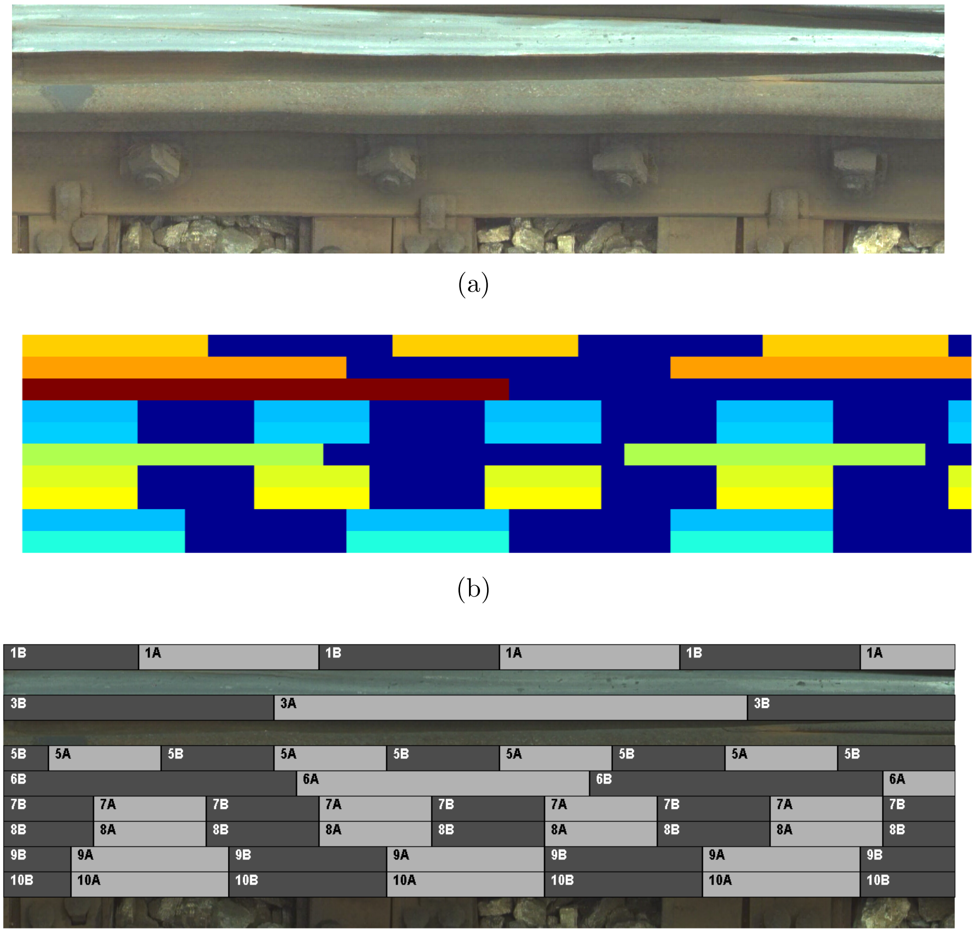
3.5 Experimental Results

Components are detected in panoramic images of railroad track. Video data was acquired from a camera on a hand-pushed cart that captured video at 30 frames per second. From the video, we created a panoramic image by first estimating the interframe displacement, then stitching the video frames together using the computed displacement, as described in Section 2.3. The resulting panoramas were $N_y = 360$ pixels in height and between $N_x = 1000$ and $N_x = 2000$ pixels wide. To form the M Gabor blocks, we decomposed the image into overlapping blocks of size $N_b = 64$. The blocks overlap with their neighboring blocks by $(N/2)$ pixels, so $M = \frac{N_x}{(N_b/2)}$.

We computed T_k, $\mathbf{p}_k$, and υ_k for all components in two separate track panoramas. The first panorama contained sequential bolts and the second panorama contained track components that were acquired from the lateral viewpoint described in Section 2.4.1.

The panorama in Figure 3.3(a), contains four consecutive bolts. Figure 3.3(b) indicates the detected periodicity for each row $I_r(x)$. Color is proportional to the SNR of the detected T_k, ranging from strongest (red/orange) to weakest (light blue). This is also apparent in gray scale, where the brightest rows correspond to the highest SNR. The patterns that were chosen for each row are proportional to the detected period T_k, but are illustrated without regard to υ_k, so their positive and negative duty cycles do not align with any particular object in the horizontal direction. High frequency is detected in the rows containing the bolts. Figure 3.3(c) shows the consecutively labeled rows, where A arbitrarily indicates components of one type, and B indicates components of another type. In some experiments, A and B were both periodic components, and in others only one was periodic. In Figure 3.3(c), υ_k has been learned from the algorithm, so the boundaries of each component were drawn accordingly.

The resulting detected components are shown in Figure 3.4. Note that the original image is not exactly periodic. The spacing between the bolts is not


1B 1A 1B 1A 1B 1A
3B 3A 3B
5B 5A 5B 5A 5B 5A 5B 5A 5B
6B 6A 6B 6A
7B 7A 7B 7A 7B 7A 7B 7A 7B
8B 8A 8B 8A 8B 8A 8B 8A 8B
9B 9A 9B 9A 9B 9A 9B
10B 10A 10B 10A 10B 10A 10B

(a)

(b)

(c)

Figure 3.3: Track bolts. (a) Panoramic image of inspected track with periodic bolts. (b) Periodicity T_k per row, where image color is proportional to SNR. (c) Labeling of each row using the detected period, T_k, and phase shift, υ_k.

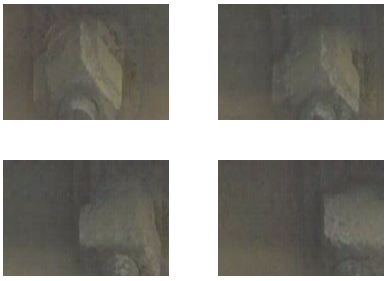
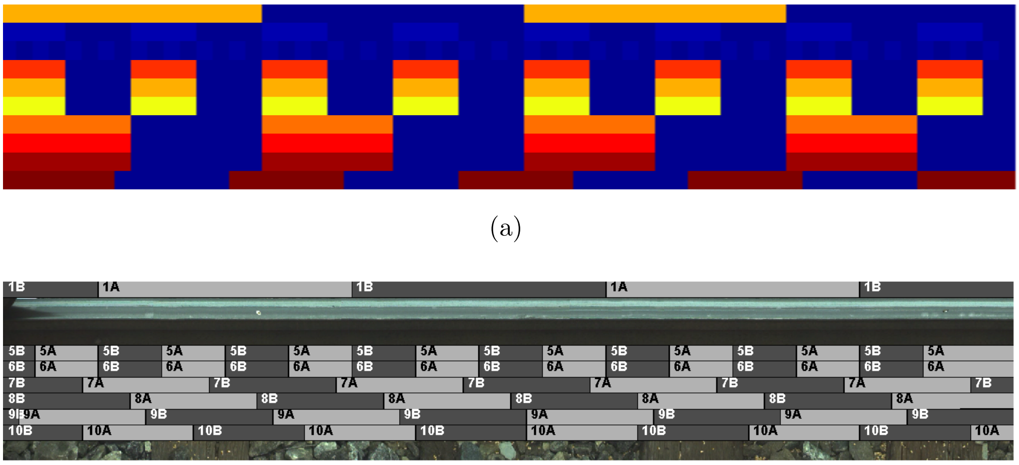


Figure 3.4: Detected bolts (Object 7A).



(a)

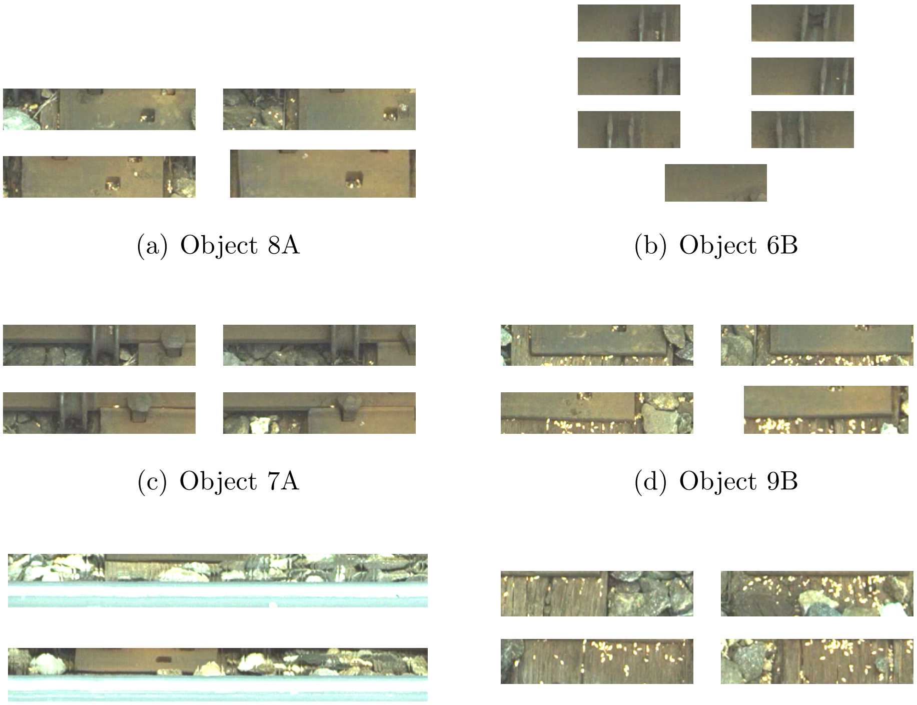
(b)

Figure 3.5: Periodicity detection of track panorama from Figure 3.1. (a) Periodicity T_k per row, where image color is proportional to SNR. (b) Labeling of each row using the detected period, T_k, and phase shift, υ_k.

exactly uniform, yet our algorithm chooses a best-fit period that contains much of the bolt component, as shown in Figure 3.4. We refer to the bolts as Object 7A because they are located at the $7A$ labeling in in Figure 3.3 (c).

The second panorama that we examined was shown in Figure 3.1. Ten groups of image rows, $I_1(x), I_2(x), \ldots, I_{10}(x)$, are processed. The results of the initial T_k estimates are shown in Figure 3.5(a), where again the colors/illumination are proportional to SNR. From these initial periodicity T_k estimates, we solve for υ_k according to our algorithm. The detected components' T_k and υ_k can be seen in Figure 3.5(b).

Detected components are shown in Figure 3.6. The tie plates are effectively located as Object 8A in Figure 3.6(a). Note that in the original image, Figure 3.1, the anchors are missing from the rightmost tie. Our algorithm worked well despite this, and in components such as 6B and 7A (Figures 3.6(b) and 3.6(c)), it is evident that the missing anchor causes the components to not look identical. Object 9B is shown in Figure 3.6(d). Our method is able to separate ballast texture from tie texture, despite the presence of unknown granular material on the tie. Periodicity can be detected in the first rows of the panorama (Object 1A in Figure 3.6) even though the image itself has poor quality at that location. In Figure 3.6(f), it is again demonstrated that the tie texture can be detected even in the presence of anomalous environmental



(a) Object 8A

(b) Object 6B

(c) Object 7A

(d) Object 9B

(e) Object 1A

(f) Object 10B

Figure 3.6: Detected primary components.

elements.

We continued to process the panorama in Figure 3.1 to detect secondary periodicities. We use the MUSIC algorithm to find the next dominant frequency, as described in Section 3.4.3. The results are shown in Figure 3.7, where Rows 6, 7, and 10 all contain strong secondary periodicities. Note that Row 7, which contained a lower-frequency tie plate component in Figures 3.5 and 3.6(c), now produces a high frequency component corresponding to the anchor. Also note that Object 7A in Figure 3.8(b) includes images of both the top of a tie plate and of ballast. This is correct, since an anchor occurs either to the left or right of a tie plate, so Object 7A does not occur with the same periodicity as object 7B. Similarly, Row 6, which had contained a higher-frequency anchor component in Figure 3.6(b), now contains a lower-frequency tie plate component, as shown in Figure 3.8(c). When processing Row 10, it is evident from the original image, Figure 3.1, that there is both an aberration of the periodicity due to the left-most tie being skewed, and also due to heavy ballast cover on the ties. Nevertheless, our algorithm found the best approximation to the primary component in Figure 3.6(f). A lower frequency is detected in Figure 3.8(d), that contains both tie and neighboring ballast as one unit.

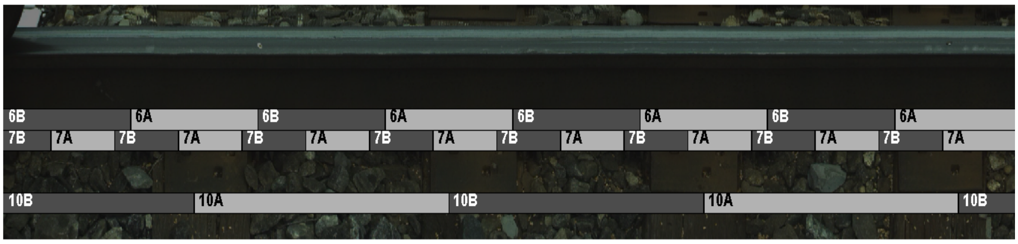
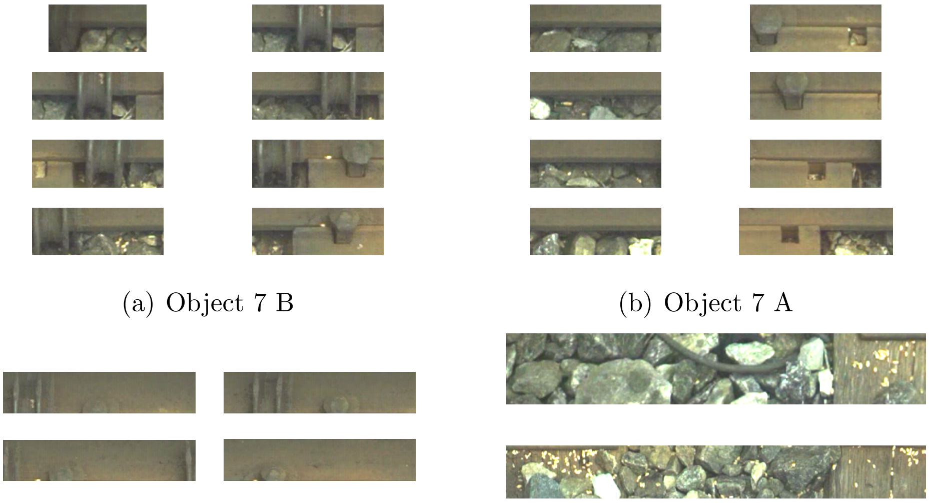


Figure 3.7: $I_r(x)$ with strongest secondary periodicities ($r = 6$, 7, and 10).



(a) Object 7 B

(b) Object 7 A

(c) Object 6 A

(d) Object 10 B

Figure 3.8: Detected secondary components.

The algorithm yielded satisfactory results in our experiments. It was robust to periodicity aberrations and anomalous appearances. The detected T_k and υ_k reliably delineated many of the periodically occurring components.

3.6 Turnout Inspection

Our periodic component detection method is also useful for turnout detection. As Section 2.2.1 stated, the turnout is a vital part of the track that must be carefully inspected. First, however, it must be detected that a turnout is being inspected, as the components there are quite different than in regular straight or curved track. We detect the presence of a turnout using the lateral viewpoint.

We created a signal processing-based method for detecting periodic components indicative of turnouts, such as frog bolts or joint bar bolts (Figure 3.9), and estimating that period, T. The estimation of periodic component location within turnouts is carried out by converting the video of the middle portion of the video, containing the rail web, into a panoramic mosaic (Figure 3.10(a)). The periodicity of the components in the panoramic mosaic are then estimated, and the components subsequently localized. We utilize a block-wise Gabor transform for computational efficiency, since the expected sizes of the components are known.

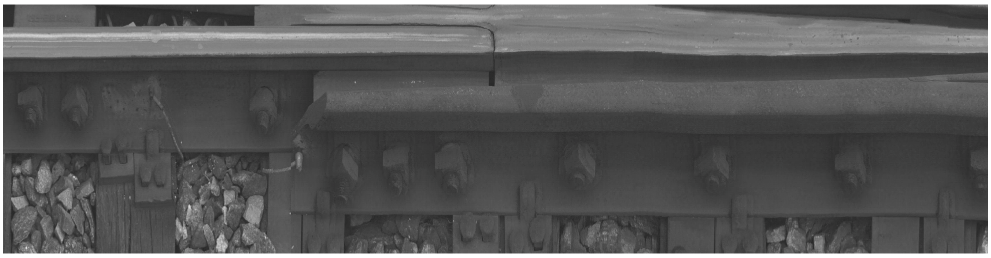
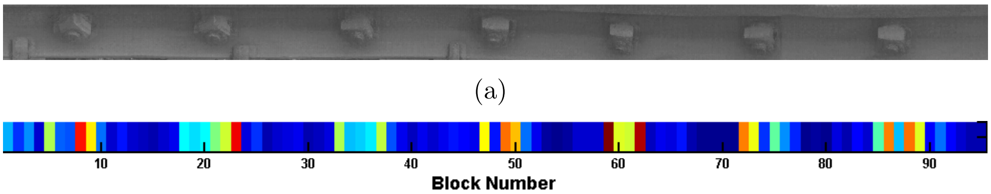


Figure 3.9: Original turnout panorama.

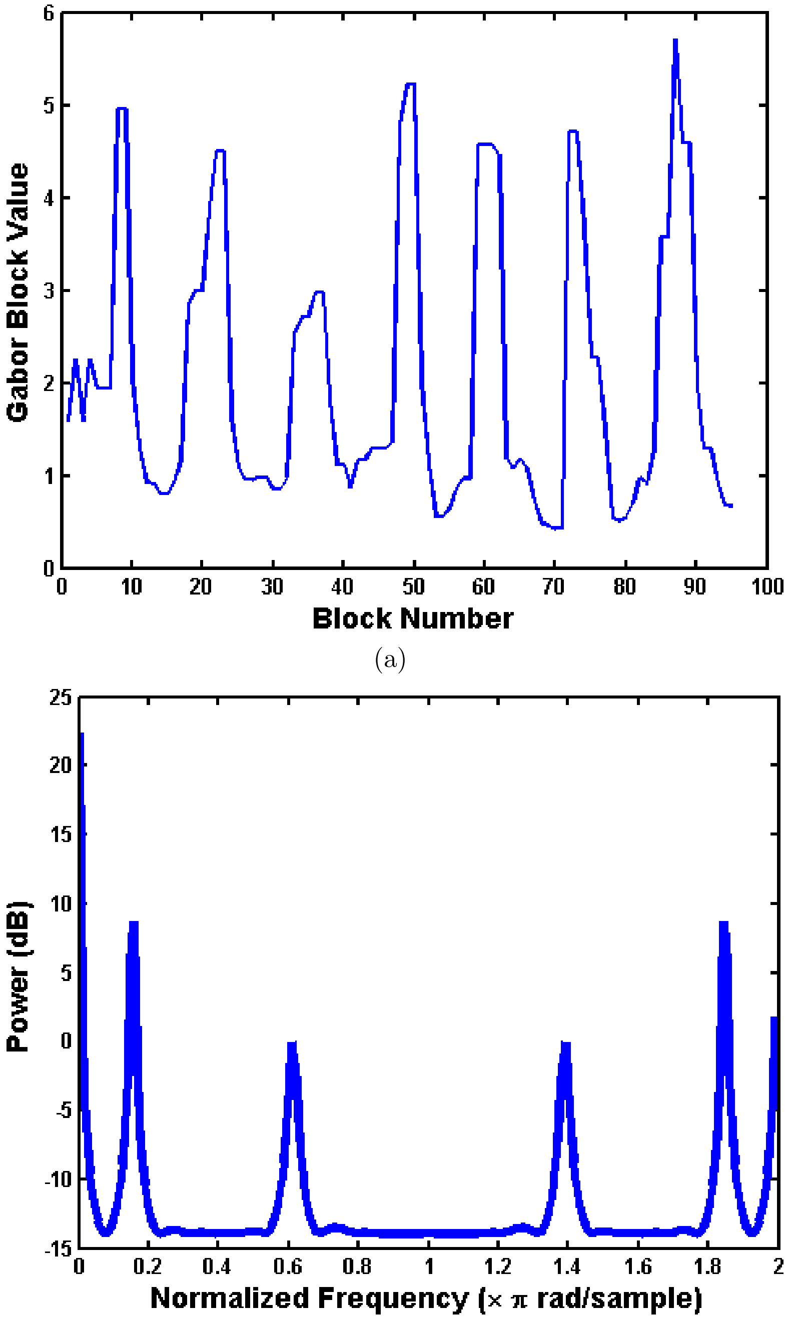


(a)

(b)

Figure 3.10: Turnout bolts used in recognition. (a) Original panoramic image and (b) block-wise Gabor image.

The image is transformed in a block-wise manner into the Gabor frequency domain (Figure 3.10(b)). Each block's height is identical to the height of the rail web area shown in Figure 3.10(a), and each block's response is computed using an overlapping window with respect to each block's right neighbor. This window overlaps by half of the block's width. This block-wise Gabor response is then processed as a one-dimensional signal (Figure 3.11(a)). The



(a)

(b)

Figure 3.11: Turnout detection: (a) $\mathcal{G}(I_r(x))$ and (b) $J(\omega)$ of $\mathcal{G}(I_r(x))$.

MUSIC algorithm is subsequently applied to find periodic components (Figure 3.11(b)).

The MUSIC algorithm outputs a frequency analysis, in which the input signal's frequency response is computed for each frequency (Figure 3.11(b)). Dominant frequencies are then detected. The output of 3.11(b) shows the power at each radial frequency, ω. Each radial frequency relates to the period, T, by the formula $\omega = \frac{2\pi}{T}$. Hence, when the peak is located at $T = 0.14$, the component repeats every $T = 14.3$ blocks. This is a satisfactory approximation for the bolts in Figure 3.9, and this distance can vary depending on the turnout angle, component and turnout design, and turnout manufacturer.

If localization is needed, then autocorrelation can be performed on the blocks in the Gabor frequency domain. Candidate blocks would be proposed that have a strong Gabor frequency response (Figure 3.10(b)). The autocorrelation between a candidate block and all blocks that are nT blocks apart would be measured, where n is a positive integer. Blocks that yield a strong Gabor response and that are highly correlated to blocks nT away would be identified as repeating components.

3.7 Conclusion

This chapter presented a detection and localization algorithm for inspection panoramas that contain periodically repeating parts in one direction. The algorithm that we presented relies on a panoramic image, which was obtained by concatenating parts of successive video frames from a field-acquired inspection video. This algorithm operates in an unsupervised manner, but with the prior knowledge that all of the periodically repeating components occur in the horizontal direction. Since it was unsupervised, however, the components that were detected may or may not have been the components that a human would have identified.

Such an algorithm has valuable future applications to railroads for more extensive and reliable monitoring than is currently feasible. The use of this periodicity detection algorithm, which can identify specific sections of track based on the appearance of periodic component locations, will be key to invoking distinct machine vision algorithms to identify and inspect the particular components normally found in track sections such as turnouts and other special track work. This was demonstrated with the turnout detection. By detecting a turnout this way, as opposed to GPS, we already have isolated key components, such as turnout switch rod bolts, so that inspection can proceed quickly.

The one-dimensional algorithm can be further developed to provide ways to adapt to changing environmental conditions. The results of the one-dimensional algorithm are encouraging, and this work naturally has a two-dimension extension. Because MUSIC is not designed for multiple dimensions, the extension to two dimensions is non-trivial. Additionally, making an algorithm general to all sizes, orientations, and appearances of objects is difficult. A two-dimensional algorithm is created in Chapter 4.

CHAPTER 4

TWO-DIMENSIONAL PERIODICITY

Chapter 3 demonstrated the effectiveness of periodicity detection for an inspection panorama where the periodicity was present in a single direction. As a further simplification, the choice of Gabor block size N was given. This chapter explores the more general case, where periodicity may occur in two dimensions, and it is unknown what choice of N is optimal for detecting the periodic object.

We extend the one-dimensional scenario from Chapter 3 to a more general two-dimensional periodicity detection. The two-dimensional image is Gabor-transformed in a continuous, rather than block-wise, manner. The Gabor directions which yield the minimum entropy in the filtering are examined for periodicity. A method similar to the Fourier slice theorem is used to detect the angle of periodicity. This model is refined using spatial and frequency information to localize the periodically repeating components in an image.

4.1 Background

Detecting and modeling periodic textures is a well-studied problem. It is frequently studied in structural texture analysis, since many textures can be characterized by distinct periodic patterns (e.g. bricks in a brick wall). Examples of this research area include [16], [17], and [18]. Periodic textures in research, however, typically consist of tessellations of a regular structure. We detect periodicity in complex objects that may contain several textures, and that are similar but not identical.

The Fourier domain can detect some spatially periodic textures [12]. Spatial frequency can be used as a self-filter, where the highest magnitude spatial frequencies from the image itself are used in filtering, thus emphasizing the spatial frequencies that contain a dominant periodicity. However, this method falls short when used on the real world image of the track panorama. There are two reasons for this: first, the tie and ballast textures are not well-structured. Thus, quantifying them as periodically repeating tessellations is inaccurate. Second, the spatial frequencies in both the tie and ballast have similar magnitudes, and only larger gradients benefit from self-filtering (e.g. rail-to-ballast).

Wold modeling defines each image as a periodic component, a linear component, and noise [19].

Some work has been done in the field of detecting periodicity of structural objects [20]. In this paper, features of interest are first extracted. Then, filtering is created using the Fourier transform of those features convolved with the Fourier transform of autocorrelation. This is repeated iteratively, where the Fourier transform of the features is updated using the new estimated periodic estimate. Although few thresholds are used, the reliance of this method on the initialization of textures of interest makes it susceptible to noise.

Spectral estimation has been used to detect rotational and reflectional symmetry in images [21]. In [21], angular correlation is used to obtain a one-dimensional representation of an object in an image, and then spectral estimation is used to recover the periodicity, which is related to the order of symmetry. We have similarly chosen to use spectral estimation for its ability to efficiently and robustly detect the frequency in a one-dimensional signal.

4.2 Model

A two-dimensional image $I(\vec{x})$, or $I(x, y)$, consists of periodically repeating objects and non-periodic image parts. Each object that repeats does so in a direction, θ, with a periodicity of T pixels. The values for θ and T are not known a priori. Neither is spatial information such as the size and appearance of the object.

The direction of periodicity, θ, is the angle along which repeating objects will occur every T pixels. Examples of two values of θ are shown in Figure 4.1. Specifically, $\theta = 0$ and $\theta = \frac{\pi}{4}$ are demonstrated.

The proposed method will detect the periodic objects, solve for their θ and T, and extract all occurrences of those objects. Since the objects are similar

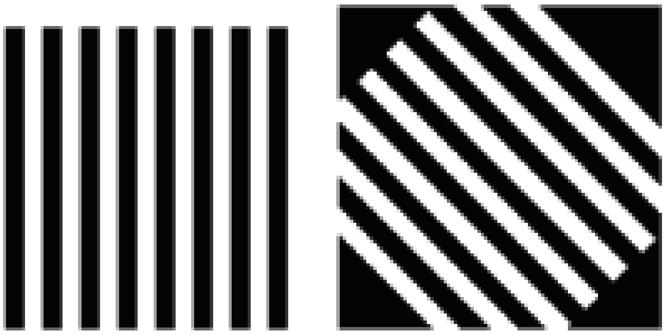


Figure 4.1: Example images for $\theta = 0$ (left) and $\theta = \frac{3\pi}{4}$ (right).

but not identical, we found that periodicity detection is most robust when $I(\vec{x})$ is first spatially filtered by many Gabor filters. Periodicity detection is then done on each of the filtered images, and the results are combined to form a single hypothesis for θ and T.

Decomposing $I(\vec{x})$ into Gabor responses helps remove spatial variations between objects, as most objects share a general shape, even if their appearance varies. It also provides several different input signals to our algorithm.

4.2.1 Gabor Filter

Each Gabor-filtered version of the image $I(\vec{x})$ is produced by a unique Gabor filter. The Gabor filters are specified by orientation (ϕ), wavelength (λ), phase offset (ψ), and spectral aspect ratio (γ) [22]. Each two-dimensional Gabor filter $g_{\lambda,\phi,\psi,\sigma,\gamma}(x, y)$ is defined as

$$g_{\lambda,\phi,\psi,\sigma,\gamma}(x, y) = \exp\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right)\cos\left(2\pi\frac{x'}{\lambda} + \psi\right) \tag{4.1}$$

where

$$\begin{aligned} x' &= x\cos\phi + y\sin\phi \\ y' &= -x\sin\phi + y\cos\phi \end{aligned} \tag{4.2}$$

We set $\gamma = 1$ and $\psi = 0$. We introduced a new variable, B, which is the size of the filter in pixels. We set $\sigma = \frac{B}{6}$ and $\lambda = \frac{2B}{5}$. This was determined empirically, and a variety of settings could achieve similar results. We varied the orientation (ϕ) from 0 to π, and we varied B from 8 to 40 pixels. The longest dimension of any image $I(\vec{x})$ was 256 pixels. Our Gabor filters are denoted as $g_{\phi,B}(x,y)$.

The Gabor transform of image $I(x,y)$, for a given filter $g_{\phi,B}(x,y)$ is denoted as $G_{\phi,B}(x,y)$.

$$G_{\phi,B}(x,y) = \sum_{\tau_x,\tau_y} I(\tau_x,\tau_y) g_{\phi,B}(x-\tau_x, y-\tau_y) \tag{4.3}$$

We create these Gabor-filtered images, and for each image we determine if the orientation ϕ is periodic in the direction θ, as measured by a formula that will be presented in Section 4.3. Two examples of the Gabor orientations $\phi = 0$ and $\phi = \frac{3\pi}{4}$ are shown in Figure 4.2.

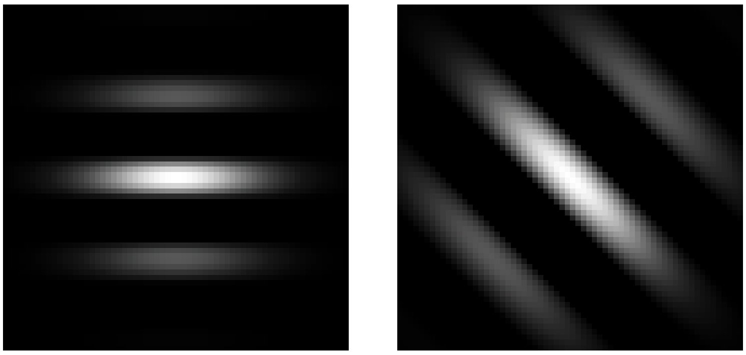


Figure 4.2: Gabor filters $g_{(\phi=0,B=64)}(\vec{x})$ (left) and $g_{(\phi=\frac{3\pi}{4},B=64)}(\vec{x})$ (right).

4.3 Periodicity Detection

Our methodology consists of periodicity detection and extraction of object occurrences. We detect the object's periodicity by analyzing all (ϕ,θ) across several scales of B with a spectral estimation algorithm. The Fourier-based methods that we use, including the spectral estimation algorithm, allow us to iteratively detect and localize several periodic objects.

Detection is challenging when there is no prior knowledge of the appearance of the object, or the direction of the periodicity. For each $G_{\phi,B}(\vec{x})$, a one-dimensional projection, $s_{\phi,B,\theta}(\lambda)$, is computed in the θ direction. Spectral estimation is then used to analyze $s_{\phi,B,\theta}(\lambda)$ and detect the periodicity T. The projection $s_{\phi,B,\theta}(\lambda)$ is defined as

$$s_{\phi,B,\theta}(\lambda) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} G_{\phi,B}(x,y)\delta(\lambda - x\cos\theta - y\sin\theta)\,dx\,dy \tag{4.4}$$

The Fourier slice theorem [23] is often used when one-dimensional projection signals are being computed on an image at an angle θ. This is because the Fourier transform of the projection is equivalent to a "slice" of the Fourier transform at θ. Let $F_{\phi,B}(u,v)$ be a two-dimensional Fourier transform of filtered image $G_{\phi,B}(x,y)$. The Fourier transform of the projection signal, $\mathcal{F}\{s_{\phi,B,\theta}(t)\} = S_{\phi,B,\theta}(\omega)$, is equivalent to

$$S_{\phi,B,\theta}(\omega) = F_{\phi,B}(\omega\cos\theta, \omega\sin\theta) \tag{4.5}$$

This means that the line formed by $(u = \omega\cos\theta, v = \omega\sin\theta)$ in the Fourier transform has values that can be computed by the Fourier transform of the projection, $S_{\phi,B,\theta}(\omega)$. This is visualized as a one-dimensional "slice" of the Fourier transform at the angle θ.

T could be estimated by identifying a high-energy frequency in the Fourier domain along the "slice" at angle θ [23]. However, since we are detecting a small number of periodic signals that are superposed onto a signal, we instead use spectral estimation. Specifically, the Multiple Signal Classification (MUSIC) spectral estimation algorithm is applied to the projection signal $s_{\phi,B,\theta}(t)$ due to its ability to extract sinusoidal signals more robustly. Let $\mathbf{s} = s_{\phi,B,\theta}(t)$, the MUSIC algorithm states that the signal $\mathbf{s}$, typically referred to as a "received" signal, is modeled as

$$\mathbf{s} = \mathbf{A}\mathbf{d} + \mathbf{z} \tag{4.6}$$

where $\mathbf{A}$ is the signal subspace, $\mathbf{d}$ is the vector of signal amplitudes with respect to that subspace, and $\mathbf{z}$ is a noise vector [14]. In MUSIC, the goal is to find the sinusoidal signals $\mathbf{d}$, and this is achieved by finding the dominant frequencies on the noise projection subspace. First, the covariance matrix $\mathbf{R}_s$ is computed where

$$\mathbf{R}_s = E\{\mathbf{s}\mathbf{s}^H\} = \mathbf{A}\mathbf{R}_d\mathbf{A}^H + \sigma_z^2\mathbf{I} \tag{4.7}$$

where $\mathbf{R}_s$ and $\mathbf{R}_d$ are the autocorrelation matrices of $\mathbf{s}$ and $\mathbf{d}$ respectively, E is the expected value symbol, σ_z is the standard deviation of noise, and $\mathbf{I}$ is the identity matrix. The noise eigenvectors are then used to recover the sinusoidal signals $\mathbf{d}$ in the received signal $\mathbf{s}$. We apply the MUSIC algorithm to the signal $s_{\phi,B,\theta}(\lambda)$ to obtain an estimate for $T_{\phi,B,\theta}$.

4.3.1 Quantifying Spatial Correctness

Each estimate of $T_{\phi,B,\theta}$ is measured for correctness. We spatially verify that the periodicity of each one-dimensional projection signal is correctly estimated by measuring the strength of the alternating local minima and maxima, as they occur every $T_{\phi,B,\theta}$. For the projection signal $s_{\phi,B,\theta}(t)$, the maximum and minimum values are detected at locations $t_{\max}$ and $t_{\min}$, respectively. The quality metric, $Q_{\phi,B,\theta}$, is then computed as follows:

$$\begin{aligned}
q_1 &= \sum_{n_1}(s_{\phi,B,\theta}(t_{\min} + n_1 T_{\phi,B,\theta}) \\
&\quad - s_{\phi,B,\theta}(t_{\min} + n_1 \frac{T_{\phi,B,\theta}}{2})) \\
q_2 &= \sum_{n_2}(s_{\phi,B,\theta}(t_{\max} + n_2 \frac{T_{\phi,B,\theta}}{2})) \\
&\quad - s_{\phi,B,\theta}(t_{\max} + n_2 T_{\phi,B,\theta}) \\
Q_{\phi,B,\theta} &= q_1 + q_2
\end{aligned}$$

where n_1 and n_2 are all positive integers such that the offsets needed for the quality metric still are within the length of the signal $s_{\phi,B,\theta}(t)$. $Q_{\phi,B,\theta}$ will have a high value for signals with large amplitudes that alternate at a regular periodicity $T_{\phi,B,\theta}$. In practice, a range of values around the offsets were used, since an exact $T_{\phi,B,\theta}$ is restrictive.

4.3.2 Detecting (ϕ, θ)

$\Lambda(\phi, \theta)$ is the cumulative measure for quality across all scales B:

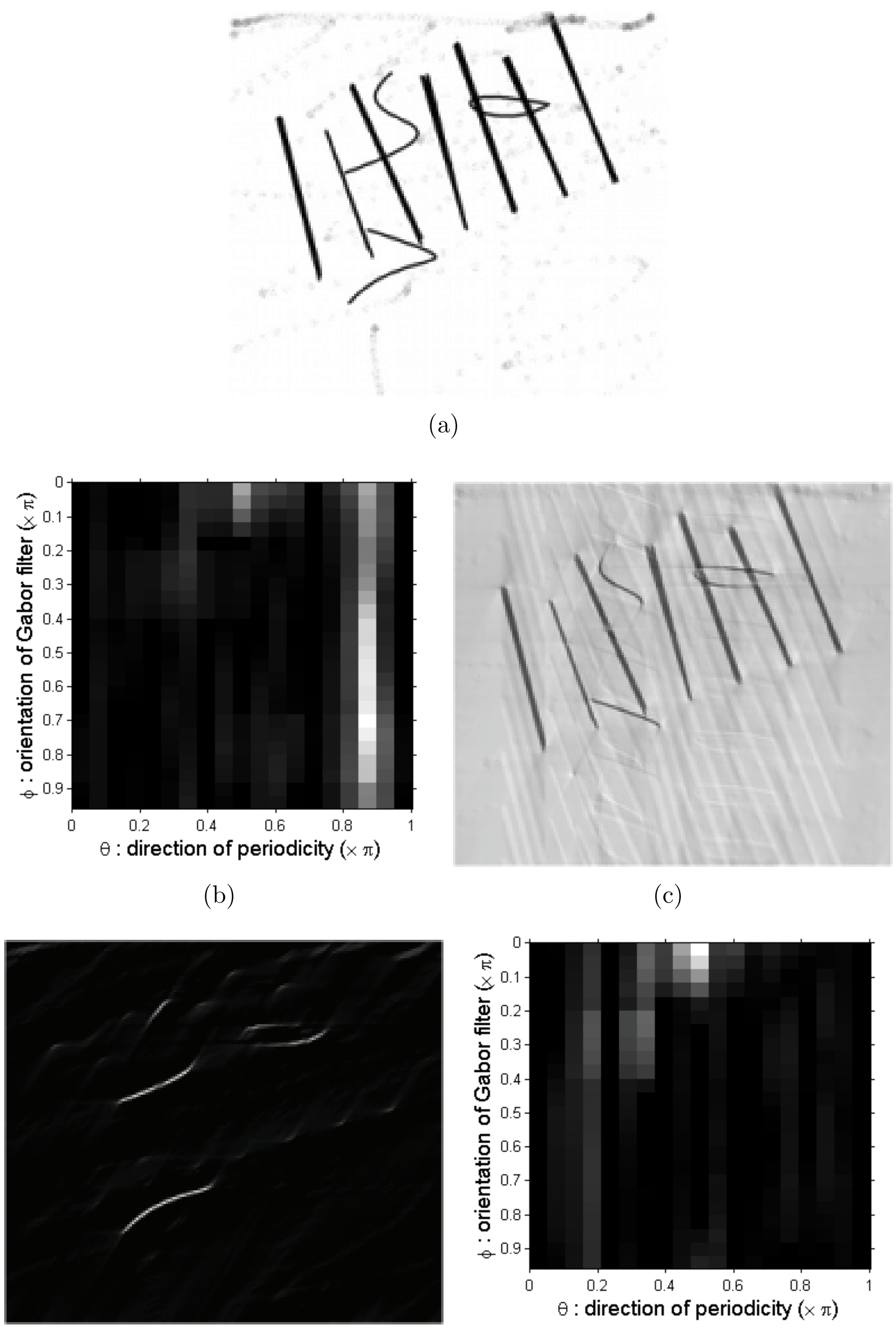
$$\Lambda(\phi, \theta) = \sum_{B} Q_{\phi,B,\theta} \tag{4.8}$$

We select the (ϕ_0, θ_0) that yields the maximum $\Lambda(\phi, \theta)$. Neighboring values of ϕ often yield very similar responses. Hence, we will use a set of ϕ, which we denote as $\mathbf{\Phi}$, in the Sections 4.4 and 4.4.1. We set $\mathbf{\Phi} = \{\phi_0\}$, and then add to $\mathbf{\Phi}$ all neighboring ϕ with a value of $\Lambda(\phi, \theta_0)$ within one standard deviation of $\Lambda(\phi_0, \theta_0)$, where the standard deviation is computed from the set of all $\Lambda(\phi, \theta_0)$.

4.4 Iteration

After the orientation θ_0 that produces a dominant periodicity is detected, other periodic objects should also be detected. The Fourier slice theorem, which was our motivation for using spectral estimation on $s_{\phi,B,\theta}(t)$ in Section 4.3, is used iteratively to suppress dominant periodic objects so that objects with a less-dominant periodicity can be discovered. One can suppress the "slice" corresponding to ϕ_0 in the Fourier domain and reconstruct the object. That is, set $F(\omega \cos\phi_0, \omega \sin\phi_0) = 0$ where $F(u, v)$ is the Fourier transform of the image $I(\vec{x})$. An example is shown in Figures 4.3(a)-(e). The image in Figure 4.3(c) is reconstructed only from the image "slice" $F(\omega \cos\phi_0, \omega \sin\phi_0)$. The residual image shown in Figure 4.3(d) is formed by suppressing $F(\omega \cos\phi_0, \omega \sin\phi_0)$ in the Fourier domain and then transforming the Fourier domain back to the spatial domain. Notice how the horizontal lines are left intact in Figure 4.3(d). In fact, they contain a localized periodicity along $\theta = \frac{\pi}{2}$. Although the objects are smaller and the periodicity only occurs for two cycles, the periodicity is evident in the updated Figure 4.3(e) (note the brightness around $\phi = 0, \theta = \frac{\pi}{2}$).

The ladder example in Figures 4.4(a)-(e) demonstrates similar results for the ladder. The original $\Lambda(\phi, \theta)$ has several high values, but the highest occurs around $\theta_0 = \frac{\pi}{2}$ and $\phi_0 = \pi$. The reconstruction of $F(\omega \cos\phi_0, \omega \sin\phi_0)$ in Figure 4.4(c) shows the horizontal rungs of the ladders. The residual image of Figure 4.4(d) no longer has rungs, and it is now evident that there is some approximately horizontal periodicity (at $\theta \approx 0$) due to the sides of the ladder and the shape of the bushes. This periodicity is not as regular, but it is



(a)

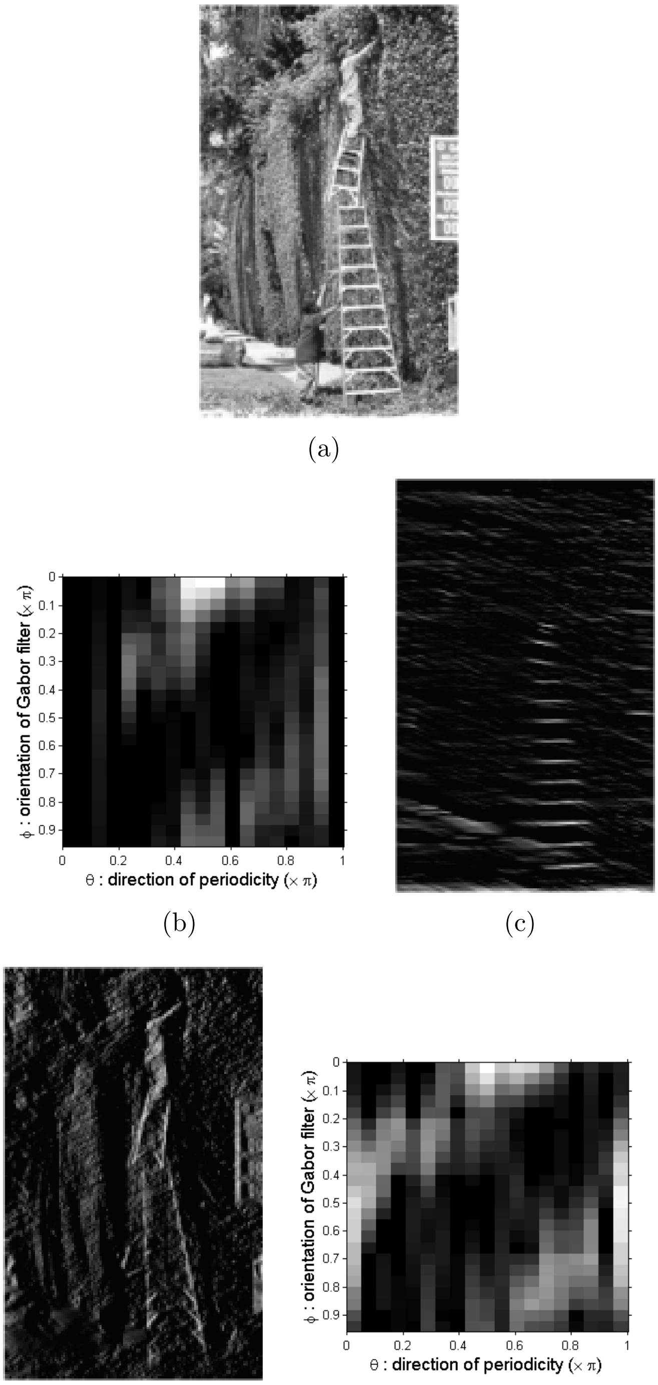
(b)

(c)

(d)

(e)

Figure 4.3: (a)-(e) Synthetic example. (a) Original synthetic image, (b) $\Lambda(\phi, \theta)$, (c) reconstructed image from $F(\omega \cos \phi_0, \omega \sin \phi_0)$, (d) reconstructed residual image where $F(\omega \cos \phi_0, \omega \sin \phi_0) = 0$, and (e) $\Lambda(\phi, \theta)$ from residual image.


(a)

(b)

(c)

(d)

(e)

Figure 4.4: (a)-(e) Ladder example. (a) Original ladder image, (b) $\Lambda(\phi, \theta)$, (c) reconstructed image from $F(\omega \cos \phi_0, \omega \sin \phi_0)$, (d) reconstructed residual image where $F(\omega \cos \phi_0, \omega \sin \phi_0) = 0$, and (e) $\Lambda(\phi, \theta)$ from residual image.

evident in the updated $\Lambda(\phi, \theta)$ in Figure 4.4(e).

This method of successively suppressing periodic objects in the Fourier domain is superior to successively removing them in the spatial domain. Periodic objects can be spatially contained within other periodic objects, and this can be difficult to detect using spatial windowing.

4.4.1 Localization

Once the objects' periodicities have been detected, we localize and segment all occurrences of the repeating objects. There are two methods to achieve this: extracting the objects as repeating edges, or extracting the objects as repeating blocks.

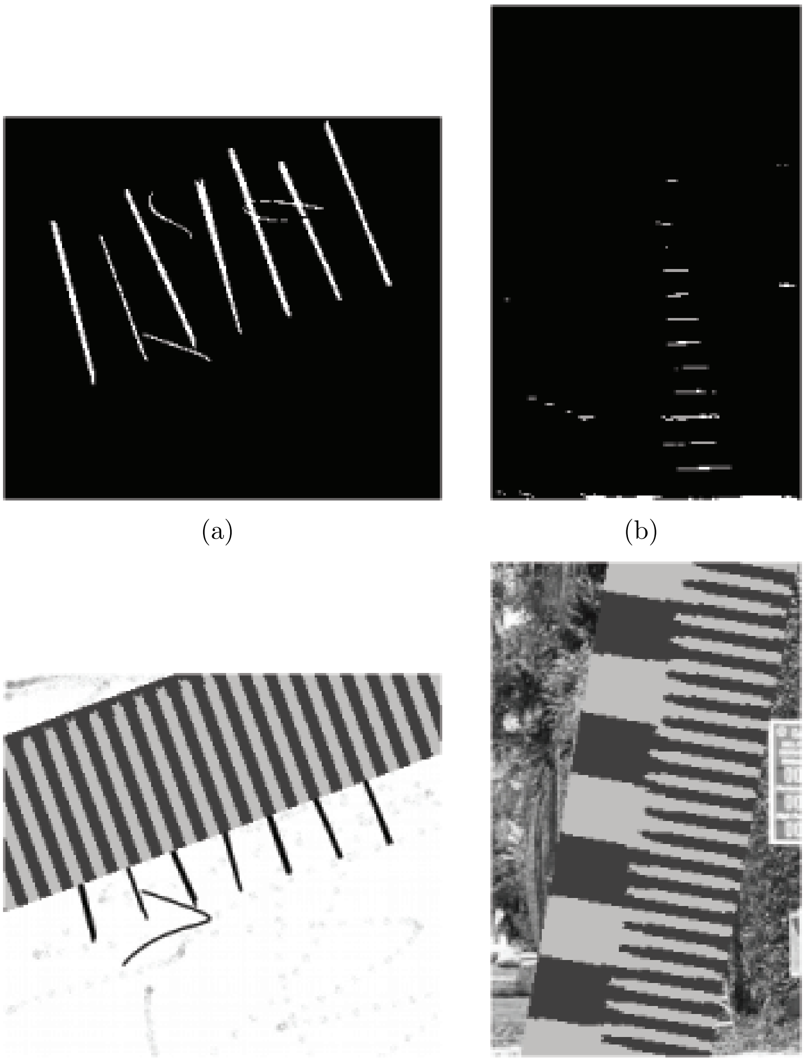
4.4.2 Edge-Based Segmentation

The Gabor features that are oriented at $\mathbf{\Phi}$ can also be spatially localized in the image by thresholding all $G_{\phi,B,\theta}(\vec{x})$, where $\phi \in \mathbf{\Phi}$, and combining the results spatially. Alternately, one can also threshold the image reconstruction from $F(\omega \cos \phi_0, \omega \sin \phi_0)$, as shown in Figures 4.4(c) and (h). Additional refinements, such as spatially filtering all edges that are not periodic with respect to neighboring edges, can also be done. The results of the edge-based segmentations that are obtained from thresholding Figures 4.4(c) and (h) are shown in Figure 4.5(a) and (b).

4.4.3 Blockwise Localization

Blockwise localization is necessary if the repeating objects are large in size and an edge does not adequately localize the objects. Both images shown in Figures 4.4 (a) and (f) contain edge-like repeating objects, but the objects can also be detected in a blockwise fashion. This is shown in Figure 4.5(c) and (d).

Periodicity is computed in a row-wise manner, where the rows are oriented in the direction θ_0. The size of these rows can be determined by finding the scale, B, that produced the maximum value of Q_{ϕ_0,B,θ_0} for $\Lambda(\phi_0, \theta_0)$. We



(a)

(b)

(c)

(d)

Figure 4.5: (a) Edge-based segmentation of synthetic image, (b) edge-based segmentation of ladder image, (c) blockwise segmentation of synthetic image, and (d) blockwise segmentation of ladder image.

denote this as B_0. In a row-wise manner, periodic objects are detected and localized.

Spectral estimation, like other Fourier-based algorithms, has high precision in the frequency domain, but no spatial resolution. We develop a method to localize the object which assumes that all objects repeat with a 50% duty cycle. That is, if an object occurs every T pixels, the component itself is $\frac{T}{2}$ pixels long.

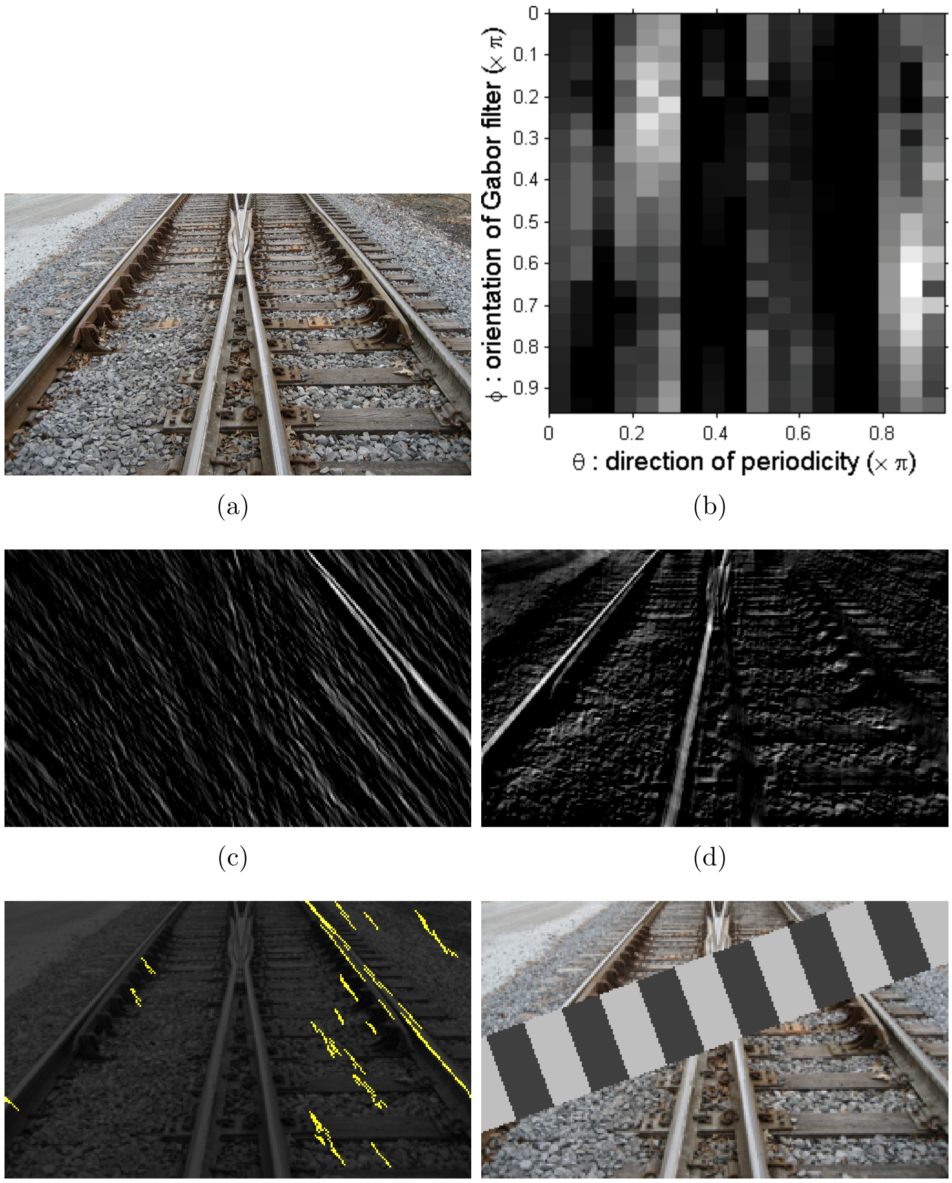
Since we are operating in a row-wise manner, we transform each row into a one-dimensional projection signal. After θ_0 is detected, as described in Section 4.3, we define a signal $\mathbf{w} = w_{B_0,\phi}(t)$, where

$$w_{B_0,\phi}(t) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} G_{\phi,B_0}(x,y)\delta(x - t\cos\theta_0, y - t\sin\theta_0)\, dx\, dy \tag{4.9}$$

We can follow the reweighting method of Section 3.4.1 for the values of ϕ. Initially, all $\alpha_\phi(x) = 1$. For all x contained in each row, we hypothesize that $x = x'$ is an object and set $\alpha_\phi(x') = G_{\phi,B_0}(x')$. MUSIC provides an SNR for the periodicity that it detects, so for every candidate x', we obtain an SNR value. The x' that yields the highest SNR according to MUSIC is considered our "object," and the duty cycle is produced such that the object is contained in a single cycle.

4.5 Experimental Results

Figures 4.6, 4.7, and 4.8 demonstrate the complete iterative process for detecting and segmenting periodic objects in the original image, 4.6(a). In each of the iterations, the image that is currently being examined for periodic objects is shown in Figures 4.6(a), 4.7(a), and 4.8(a), respectively. The measure of $\Lambda(\phi, \theta)$ for each of the images is shown in Figures 4.6(b), 4.7(b), and 4.8(b), respectively. From this measure of $\Lambda(\phi, \theta)$ a maximum value was found, and a set of Φ was selected from this. The image was filtered by the value of ϕ. The images are shown as the filtered image, Figures 4.6(c), 4.7(c), and 4.8(c), along with the residual images, 4.6(d), 4.7(d), and 4.8(d). The residual images are used as input to the next stage in the algorithm. That is, Figures 4.6(d) and 4.7(a) are identical, as are Figures 4.7(d) and 4.8(a).



(a)

(b)

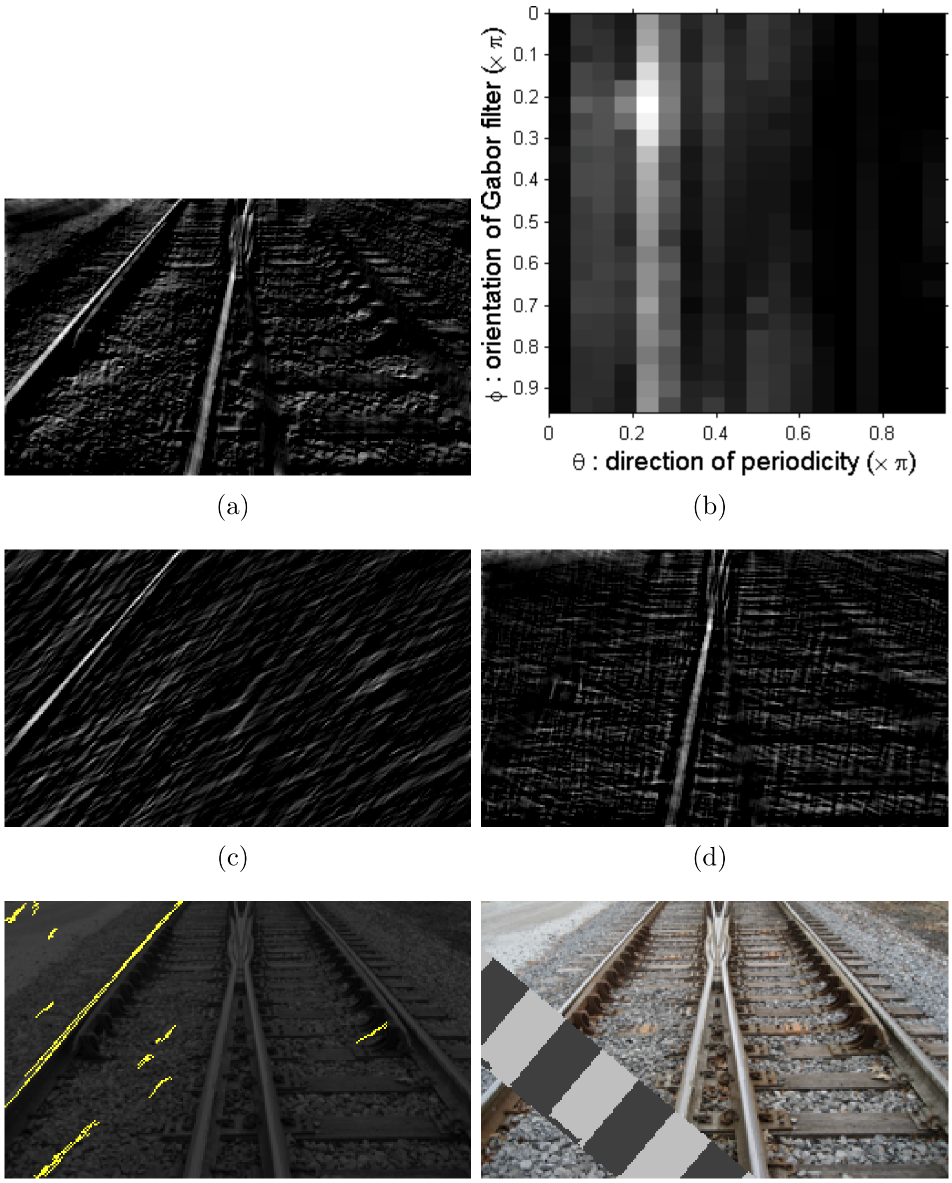
(c)

(d)

(e)

(f)

Figure 4.6: First iteration. (a) Original image, (b) $\Lambda(\phi, \theta)$, (c) reconstructed image from $F(\omega \cos \phi_0, \omega \sin \phi_0)$, (d) reconstructed residual image from $(F(\vec{\omega}) - F(\omega \cos \phi_0, \omega \sin \phi_0)$, (e) gradient-based segmentation of image, and (f) block-based segmentation of image.



(a)

(b)

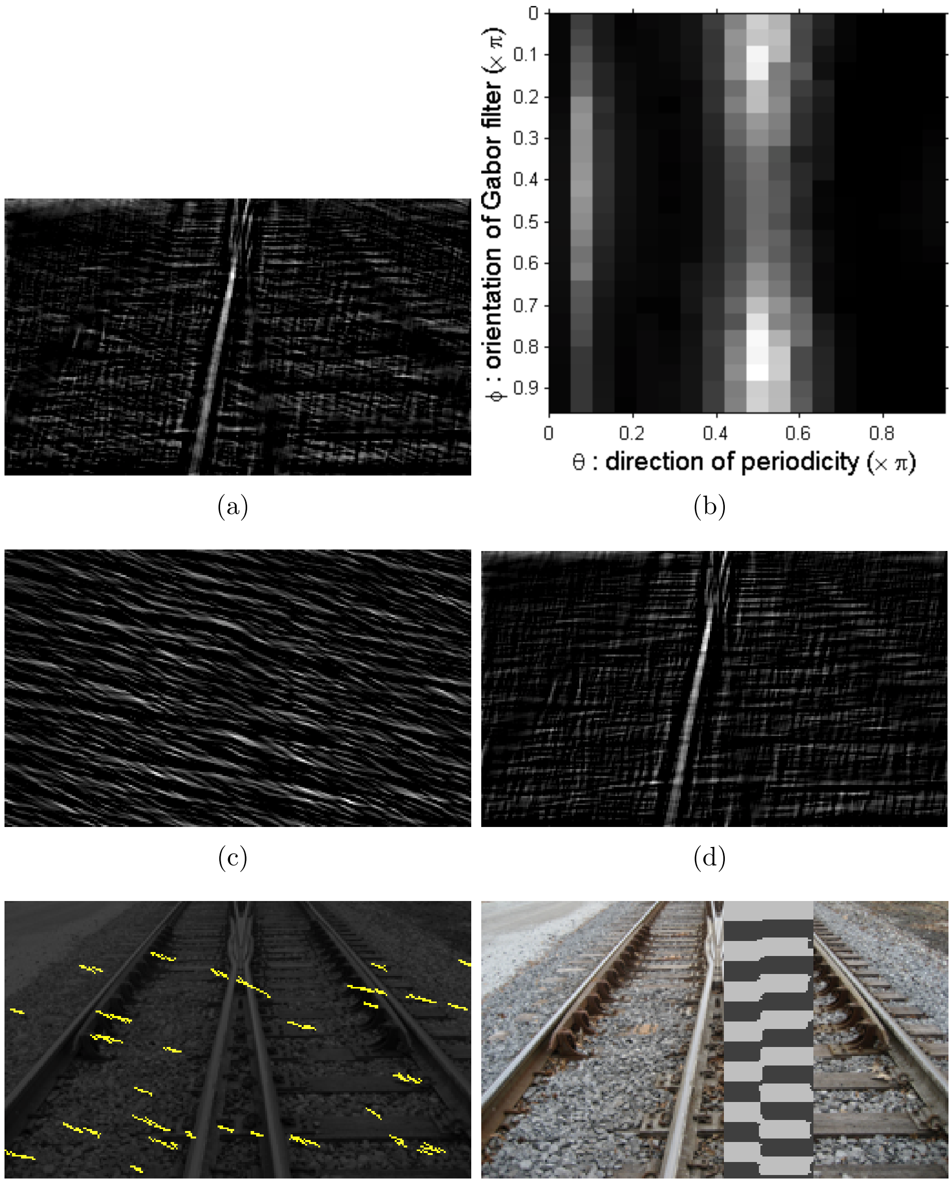
(c)

(d)

(e)

(f)

Figure 4.7: Second iteration. (a) Image from first iteration, (b) $\Lambda(\phi, \theta)$, (c) reconstructed image from $F(\omega \cos \phi_0,\, \omega \sin \phi_0)$, (d) reconstructed residual image from $(F(\vec{\omega}) - F(\omega \cos \phi_0,\, \omega \sin \phi_0)$, (e) gradient-based segmentation of image, and (f) block-based segmentation of image.



(a) (b)

(c) (d)

(e) (f)

Figure 4.8: Third iteration. (a) Image from second iteration, (b) $\Lambda(\phi, \theta)$, (c) reconstructed image from $F(\omega \cos \phi_0, \omega \sin \phi_0)$, (d) reconstructed residual image from $(F(\vec{\omega}) - F(\omega \cos \phi_0, \omega \sin \phi_0)$, (e) gradient-based segmentation of image, and (f) block-based segmentation of image.

The filtered images, Figures 4.6(c), 4.7(c), and 4.8(c), are segmented both using the gradient method and in a blockwise manner. This is shown in Figures 4.6(d)-(e), 4.7(d)-(e), and 4.8(d)-(e).

The method performs very well on this image, and the objects are segmented successfully by using the Fourier domain. This sort of separation would not have been possible in the spatial domain. Additionally, the ability to iteratively apply the method to the filtered image allows us to find all periodically occurring components.

4.6 Conclusion

A method for detecting and localizing periodic objects in images was presented. This method utilizes the robustness of spectral estimation and the spatial-frequency properties of the Gabor transform. We are able to predict the direction of periodicity using the MUSIC algorithm, and subsequently perform iterative detection as we separate periodic parts of the image from nonperiodic parts using the Fourier domain. Two methods for segmentation were presented: an edge-based method and a block-wise method. The blockwise method is used in railroad track inspection. The MUSIC algorithm detects multiple periodicities, which enables the detection of all periodically repeating components of the track. There are many applications for this method in automated track inspection, as an autonomous system will need to adapt to thousands of miles of track. Future works will include automatically updating models, utilizing the efficient Fourier-domain methods to quickly detect track defects, applying this to symmetry detection, and developing a similar method for detecting vanishing points on the horizon.

CHAPTER 5

ACTIVITY RECOGNITION

In this section, we present an activity recognition algorithm that utilizes a unified spatial-frequency model of motion to recognize large-scale differences in action using global statistics, and subsequently distinguishes between motions with similar global statistics by spatially localizing the moving objects [24]. We model the Fourier transforms of translating rigid objects in a video, since the Fourier domain inherently groups regions of the video with similar motion in high energy concentrations within its domain to make global motion detectable. Frequency-domain statistics can be used to isolate the frames that both adhere to our model and contain similar global motion; thus, we can separate activities into broader classes based on their global motion. A least-squares solution is then solved to isolate the spatially discriminative object configurations that produce similar global motion statistics. This model provides a unified framework to form concise globally-optimal spatial and motion descriptors necessary for discriminating activities. Experimental results are demonstrated on a human activity dataset.

5.1 Introduction

Activity recognition is vital to many applications including surveillance and video indexing. We propose a video model that uses a parameterized approach where one assumes that a video contains moving objects, and then attempts to extract both the motion and appearance of these objects. This is useful to a variety of applications where the domain knowledge is limited and one wishes to create a concise set of intuitive features that describe how motions vary and how their spatial configurations vary. We create a frequency-domain model that allows us to discover global motion differences between extremely different activity classes (which we refer to as meta-classes), and to localize areas where discriminative spatial configurations occur and solve for these local features. Human activity meta-classes could be humans walking versus humans staying in place while performing some action. In surveillance, global pre-processing methods that isolate video frames containing locally interesting activity are useful.

The main contributions are as follows: (1) We develop a unified spatial-frequency domain model for analyzing moving objects in a video. (2) Using the same model, we demonstrate global discrimination of meta-classes and spatial isolation of regions that produce similar global motion but have differing local properties. Finally, (3) we create a generative description for activities where spatial regions which discriminate motion classes are isolated. This section presents background, the model, and experiments to verify this approach.

5.2 Background

Activity recognition is a well-studied problem. Most work either derives global features from spatio-temporal gradients [25], or analyzes spatio-temporal cubic interest points detected using space-time gradients [26, 27]. Additionally, domain-specific algorithms have been developed that incorporate prior knowledge of the activity being performed [28]. The work of [25] provides global recognition of motion at a lower recognition rate than the state of the art using statistical operations of histograms, but with the advantages of being non-parametric and amenable to a variety of motion scenarios. Works such as [27] and [26] use spatio-temporal cubes and attempt to find cubes which are representative of a particular activity.

Although all methods achieve high levels of success on activity recognition databases, none provide a congruent method that, using one modeling framework, incorporates both global motion classification and spatial localization of moving components. Also, no other method can examine a video in its entirety and determine which frames are more immune to noise. Our work has similar advantages to many of the previous methods. We have a global scope of video motion so our features and templates are chosen with global knowledge, since energy is concentrated in the Fourier domain along the trajectory of dominant moving objects.

5.3 Frequency-Domain Signal Extraction

Our model for moving objects in video uses the frequency domain, since it is widely known to have a global scope. Recent work has shown that spatial localization can be computed if motion adheres to this model [29]. We obtain the spatial Fourier transform of each video frame at time t, $I_t(\vec{\omega})$, and model it as

$$I_t(\vec{\omega}) = \sum_{l=1}^{L} O_l(\vec{\omega}) e^{j\vec{\omega}^T \vec{\rho}_l(t)} + V_{noise}(\vec{\omega}) - V_{back}(\vec{\omega}) \tag{5.1}$$

where there are L objects, each with spatial Fourier transform $O_l(\vec{\omega})$, and each with displacement $\vec{\rho}_l(t)$ (with respect to its position at $t = 0$). For a constant velocity, $\vec{\rho}_l(t) = t\vec{v}_l(t)$. $V_{back}(\vec{\omega})$ is occluded background and $V_{noise}(\vec{\omega})$ is noise. Each frame is $N \times N$ pixels. This was described in [29]. Equation 5.1 models rigid objects, since $O_l(\vec{\omega})$ is time invariant.

Based on a technique called mu-propagation [30], setting $\vec{\omega} = (\mu_1, \mu_2)$, a signal $z_{\mu_1,\mu_2}(t)$ is introduced such that $z_{\mu_1,\mu_2}(t) = I_t(\mu_1, \mu_2)$, so that

$$\begin{aligned} z_{\mu_1,\mu_2}(t) &= \sum_{l=1}^{L} O_l(\mu_1, \mu_2) e^{j(\mu_1 t \rho_{l,x}(t) + \mu_2 t \rho_{l,y}(t))} \\ &+ V_{noise}(\mu_1, \mu_2) - V_{back}(\mu_1, \mu_2) \end{aligned} \tag{5.2}$$

where the x and y displacement of each object l is represented as $\rho_{l,x}(t)$ and $\rho_{l,y}(t)$ respectively. The terms μ_1 and μ_2 can also be expressed as $\mu_1 = \frac{2\pi m_1}{N}$ and $\mu_2 = \frac{2\pi m_2}{N}$, where (m_1, m_2) refers to a spatial frequency bin if $I_t(\vec{\omega})$ is implemented discretely.

We implement an M-length discrete short-time Fourier transform (STFT) $Z_{\mu_1,\mu_2}(p,t)$ for the signal $z_{\mu_1,\mu_2}(t)$, and now examine each STFT frequency bin $-\frac{M}{2} \leq p \leq \frac{M}{2}$ at time t. The frequency bin p is produced by the demodulator $\omega_p = \frac{2\pi p}{M}$, such that

$$|Z_{\mu_1,\mu_2}(p,t)| = |\sum_{h=0}^{H-1} z_{\mu_1,\mu_2}(t+h) w(h) e^{-j\omega_p h}| \tag{5.3}$$

where $w(h)$ is a windowing function and H is the length of that window [31]. The signal $Z_{\mu_1,\mu_2}(p,t)$ is the demodulation of $z_{\mu_1,\mu_2}(t)$ by the frequency ω_p.

Therefore, if $\vec{v}_l(t) = \vec{V}_l$ (constant-valued), then due to object l, $|Z_{\vec{\mu}}(\omega_p, t)| \propto |O_l(\vec{\mu})|$ in the frequency bin p that matches the modulating velocity $\vec{V}_l$. This occurs when $\vec{\mu}^T\vec{V}_l = \omega_p$, which will create a peak value at frequency bin $p = \frac{M}{N}\vec{m}^T\vec{V}_l$. Thus, if an object l_0 travels with $\vec{V}_l$ during the time $t+1$ to $t+h$, then Equation 6.13 becomes

$$\begin{aligned}
|Z_{\vec{\mu}}(p,t)| &= |\sum_{h=0}^{H-1} O_{l_0}(\vec{\mu})w(h) \\
&+ \sum_{h=0}^{H-1}\left(\sum_{l=1}^{L} O_l(\vec{\mu}) \prod_{k=t+1}^{t+h} e^{j(\vec{\mu}^T\vec{v}_l(k))}\right) w(h)e^{-j\omega_p h}| \\
&\approx |O_{l_0}(\vec{\mu})\sum_{h=0}^{H-1} w(h)| \qquad (5.4)
\end{aligned}$$

where the second term in Equation 5.4 is negligible if there are no other objects with velocity $\vec{v}_l$. It becomes evident that in real world applications, noise is introduced from other objects with velocities close to $\vec{v}_l$, from a time-variant $O_{l_0}(\vec{\mu})$, and from the noise terms in Equation 5.1. Figure 5.1 shows the STFT for the spatial bin $(m_1, m_2) = (0, 10)$ over time for the "galloping" sequence of the Weizmann database [32], resized with $N = 100$. Note the periodicity in the motion, due to the vertical oscillation of the person while galloping.

To use the STFT to both perform classification and identify consecutive frames that adhere to our model from Equation 5.1, we use statistics that concisely summarize the shape and the constancy of the STFT.

Spectral Centroid: The spectral centroid is a well-known function that measures the center of mass of the frequency bins as a point in time.

$$C(t) = \sum_{p=1}^{M} \frac{p|Z_{\vec{\mu}}(p,t)|}{\sum_{p=1}^{M}|Z_{\vec{\mu}}(p,t)|} \qquad (5.5)$$

Entropy: We compute the entropy across the frequency bins at each time t and create the signal

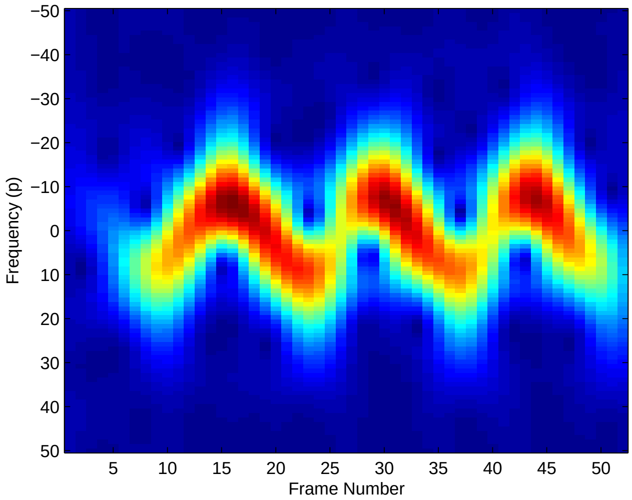


Figure 5.1: STFT of $z_{(0,0.628)}(t)$.

$$H(t) = -\sum_{p=1}^{M} \frac{|Z_{\vec{\mu}}(p,t)|}{\sum_{p=1}^{M} |Z_{\vec{\mu}}(p,t)|} log_2 \left(\frac{|Z_{\vec{\mu}}(p,t)|}{\sum_{p=1}^{M} |Z_{\vec{\mu}}(p,t)|} \right) \tag{5.6}$$

We then detect consecutive frames with $\frac{d}{dt}H(t) = 0$. A rigid object undergoing translational motion results in a constant entropy as an object with constant energy travels on a trajectory, as shown for the majority of frames in Figure 5.1. Alternately, one can look for $Z_{\vec{\mu}}(p,t)$ to be independent of time to detect consecutive frames with a constant velocity. With entropy, finding frames with $|\frac{d}{dt}H(t)| \gg 0$ is indicative of an event beyond the scope of our model (e.g. sudden appearance or disappearance of object) because a large change in entropy is indicative of a discontinuity in the phase modulation from Equation 5.1.

5.3.1 Meta-Classes

The mean value as well as the amplitude of the signals $C(t)$ and $H(t)$ are used as features to differentiate global motions. One can also use periodicity, as described in [30]. We combine classes into meta-classes so that we achieve minimal error in a linear support vector machine (SVM) which uses these statistical features. This can be replaced with a domain-specific scheme.

5.4 Spatial-Domain Template Extraction

Once we isolate frames that adhere to our model and determine meta-class membership, the spatial domain regions are solved. Each object's displacement $\vec{\rho}_l$ can be determined using only the Fourier transform of the initial video frame $I(\vec{\omega})$ and the Fourier transform of a subsequent frame $I'(\vec{\omega})$ [29]

$$\begin{aligned}\frac{I'(\vec{\omega})}{I(\vec{\omega})} &= \frac{\sum_{l=1}^{L} O_l(\vec{\omega}) e^{-j2\pi\vec{\omega}^T\vec{\rho}_l}}{\sum_{l=1}^{L} O_l(\vec{\omega})} \\ &= \sum_{l=1}^{L}\left(\frac{O_l(\vec{\omega})}{\sum_{l=1}^{L} O_l(\vec{\omega})}\right) e^{-j2\pi\vec{\omega}^T\vec{\rho}_l} \qquad (5.7)\end{aligned}$$

The values $\vec{\rho}_l$ are then determined by peak detection after an inverse Fourier transform. The frequency domain segmentation for each object, $O_l(\vec{\omega})$, is obtained using a least-squares (LS) formulation. We construct

$$Z = \begin{pmatrix} 1 & 1 & \dots & 1 \\ 1 & e^{-j\omega^T\vec{\rho}_1} & \dots & e^{-j\omega^T\vec{\rho}_L} \\ \vdots & \ddots & \vdots & \vdots \\ 1 & e^{-j(N-1)\omega^T\vec{\rho}_1} & \dots & e^{-j(N-1)\omega^T\vec{\rho}_L} \end{pmatrix} \qquad (5.8)$$

and the vector of consecutive video frames, $\vec{I} = [I_1(\vec{\omega}) \dots I_T(\vec{\omega})]$. We solve for the frequency-based motion segmentation represented by $\vec{O} = [O_1(\vec{\omega}) \dots O_L(\vec{\omega})]$ using the LS formulation

$$\vec{O} = Z^{\dagger}\vec{I} \qquad (5.9)$$

for every frequency $\vec{\omega}$. Tikhonov regularization is used to constrain the energy of $\vec{O}$ as shown in [29].

From the L frequency-based segmentations, we obtain the $N \times N$ spatial segmentation from the inverse Fourier transform of each $O_l(\vec{\omega})$. From each spatial solution, we determine the boundaries of our object from the areas of the image with the strongest gradient. To register the spatial solution with the original image, one should look for the strongest matching gradients between $I(\vec{\omega})$ and the LS solution.

5.5 Experimental Results

We demonstrate the ability of our algorithm to discriminate activity and form a generative description for activity using the Weizmann database. This database contains ten actions, each performed by nine different subjects. This database contains only one object that was necessary for activity discrimination. The signal $z_{(0,0.628)}(t)$ was created according to Equation 5.2, and statistics were created from it. $C(t)$ corresponds to vertical motion (since m_1 is set to DC). We found that the measures $|\text{median}(C(t))|$ and $(\max(C(t)) - \min(C(t)))$ make the "stationary motions" separable from the "moving motions" of this dataset (Figure 5.2). We define "moving motions" as motions where a person traverses the entire screen, while in "stationary motions" the person is not traveling.

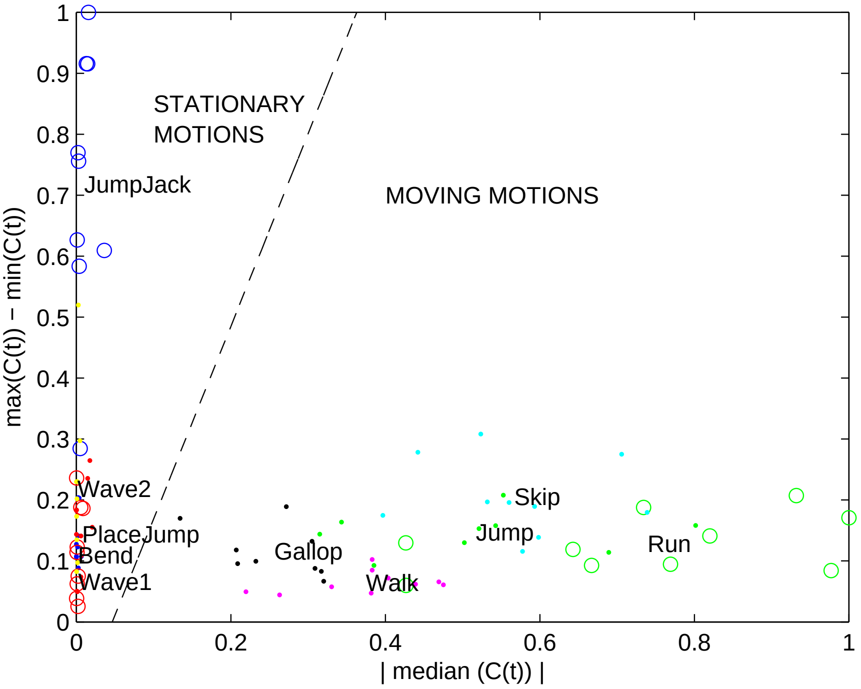


Figure 5.2: Meta-class classification.

Next we discriminate the motions within each meta-class. Figure 5.3 shows the analysis of two subjects performing a "galloping" motion as they move from right to left. We locate the maximum upward motion ($\max C(t)$) in Figures 5.3(a-b), and then find the spatial localization. The resulting objects

$\mathcal{F}^{-1}\{O_1(\vec{\omega})\}$ are shown in Figures 5.3(c-d) along with their edge-based templates. Due to human kinematics, the poses in Figures 5.3(c-d) are similar in appearance within their respective classes. In Figures 5.3(e-f), the spatial location of the objects is shown along with an arrow indicating $\vec{v}_1(t)$. Figure 5.4 shows the analysis of two subjects performing a "skip" motion as they move from right to left. We locate the maximum upward motion ($\max C(t)$) in Figures 5.4(a-c), and then spatially reconstruct the objects. The resulting objects $\mathcal{F}^{-1}\{O_1(\vec{\omega})\}$ are shown in Figures 5.4(d-f). The edge-based templates that this produces are shown in Figures 5.4(g-i).

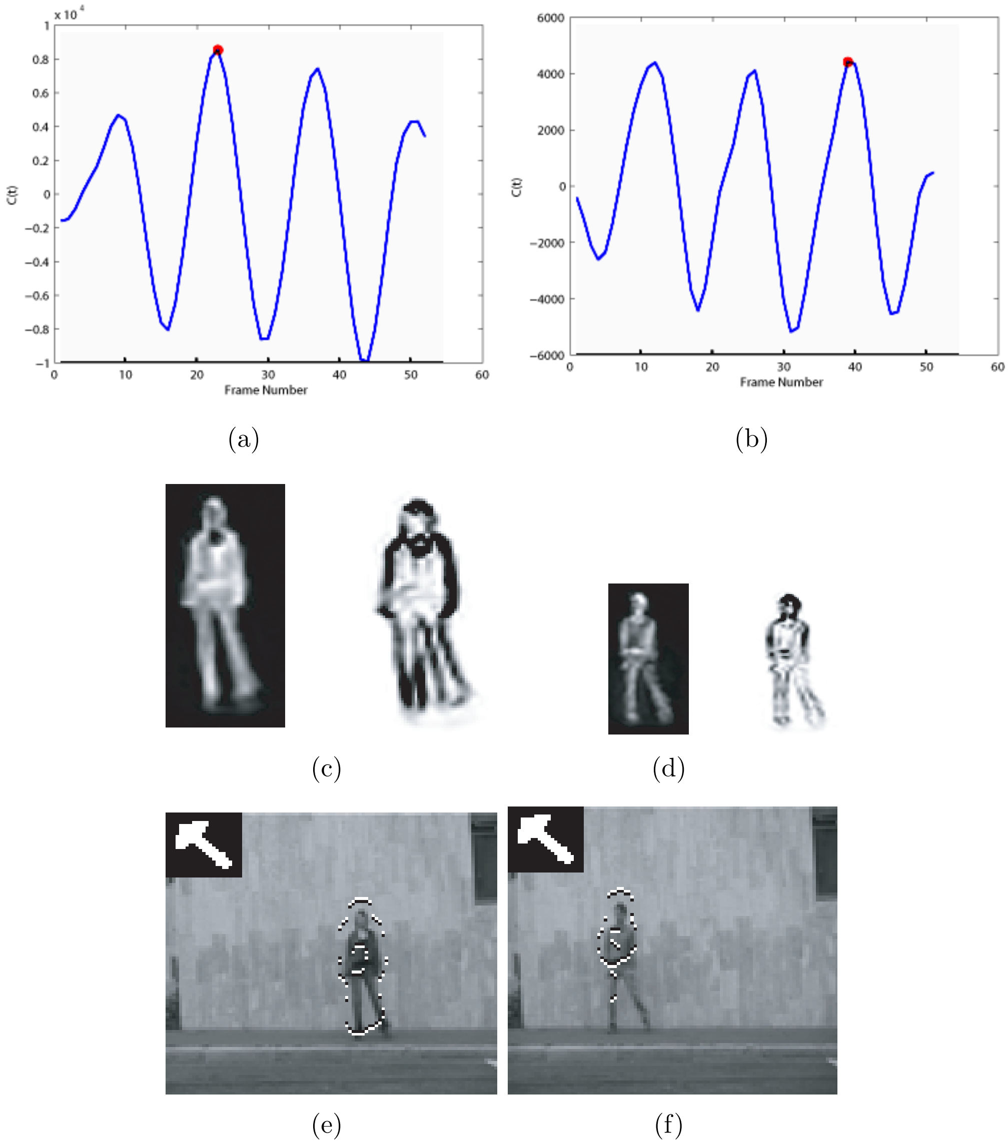
We similarly analyze the maximum downward motion ($\min C(t)$). The poses obtained during training are stored as templates, and during testing the correlation is measured between the test pose and the templates. We randomly separate the dataset into six training and three testing sequences. We average the error rates of 25 experimental runs, each with a different random permutation of the dataset.

This achieved an average recognition of above 80% on the database. Though this is below the state of the art in [26], the work here has several advantages. First, it forms a generative model which describes *how* spatial structures differ when global motion statistics are similar. This work can be extended to more complex activities in the future. Second, the global and local information that it provides is contained within the same model so that the local information is found from globally located frames of interest. Third, it allows us to locate frames globally by summarizing all of the video's content using our statistics. Spatial templates are kept to a minimum by only being formed at both the temporally and spatially discriminative areas.

5.6 Conclusion

We have provided a unified model for activity recognition that utilizes the frequency domain's ability to concentrate the energy of a moving object along the velocity trajectory in the Fourier domain. This allows both a high-level categorization of motion meta-classes and a subsequent isolation of frames that discriminate the lower sub-classes. The supporting spatial regions are then identified through a least-squares solution.

For object segmentation, a gradient-based method was effective. This is

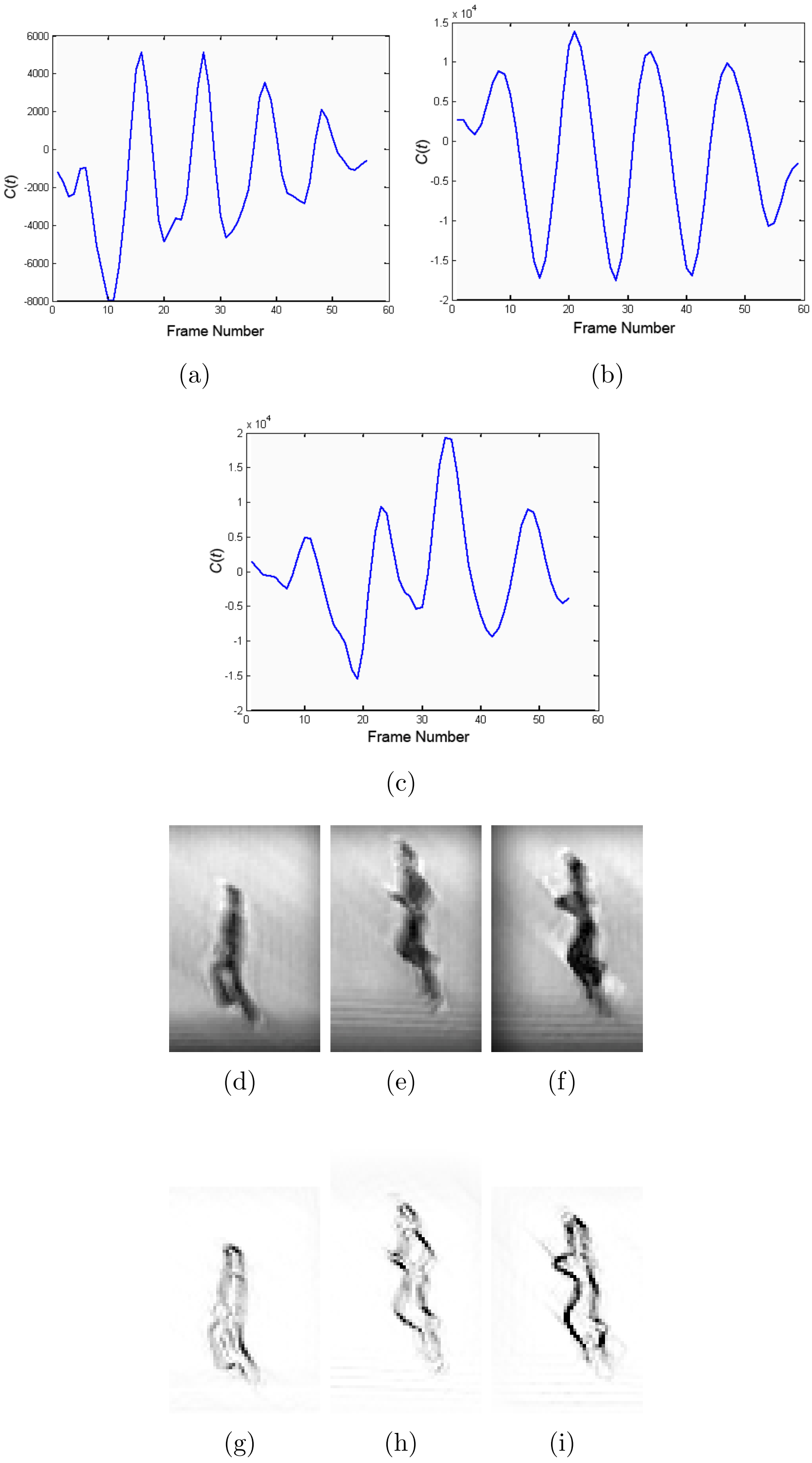


(a) (b)

(c) (d)

(e) (f)

Figure 5.3: Galloping results for two subjects. (a)-(b): $C(t)$, (c)-(d): reconstructed objects from $O_1(\omega)$ along with segmentation results using image gradients, (e)-(f): spatial location with arrow indicating $\vec{v}_1(t)$.



(a) (b)

(c)

(d) (e) (f)

(g) (h) (i)

Figure 5.4: Skip results for three subjects. (a)-(c): $C(t)$ for subjects 1, 3, and 6. (d)-(f): Reconstructed objects from $O_1(\omega)$ for subjects 1, 3 and 6. (g)-(i): Segmentation results using image gradients.

similar to the gradient-based method in Chapter 4, and it is applicable for similar reasons. The gradients that are orthogonal to the direction of motion are the ones that are reconstructed with the highest quality. Similarly, the strong gradients in Chapter 4 provide the most evidence of periodicity. This and other similarities occur because detecting periodic activity in a video is similar to detecting periodic objects in images. This is explored in great detail in Chapter 6.

CHAPTER 6

GENERAL MODEL FOR PERIODICITY DETECTION IN IMAGES AND VIDEO

In Chapter 4, periodically occurring objects were detected in images, and all occurrences of these objects were subsequently segmented. In Chapter 5, periodically occurring activities were detected in video, and the objects producing those activities were segmented. In both situations, the periodic objects and activities were similar, but not identical, to each other.

The solutions presented in Chapters 4 and 5 both utilized global information provided by the Fourier domain to detect the presence of a periodic object and to determine that object's frequency. Since the Fourier domain is unable to spatially localize an object, however, the subsequent segmentation in each algorithm relies on an edge-based or block-based segmentation method.

In Chapters 4 and 5, the image and video being processed were separated into one-dimensional signals. In Chapter 4, Gabor-filtered versions of the original image were created. For each Gabor-filtered image, one-dimensional signals were created by projecting those Gabor-filtered images along candidate directions of periodicity. In Chapter 5, video containing periodic activity was decomposed into spatial frequency components in the Fourier domain, and the instantaneous frequencies of several spatial frequency components were examined over time using the STFT. The spectral centroid of the instantaneous frequency was calculated so that a one-dimensional signal resulted from the instantaneous frequency analysis, and periodicity was detected using this signal.

In both of these applications, the two-dimensional (2D) images and the 2-D plus time (2D+t) videos are transformed into a set of one-dimensional signals. The resulting one-dimensional signals are analyzed using spectral estimation to determine their frequency. Many of these signals are orthogonal. For example, Gabor filters such as those used in Chapter 4 are orthogonal when oriented at different angles, and the spatial frequencies that are used in Chapter 5, are orthogonal. Some of these one-dimensional signals will contain periodic information and some will not; thus, our goal is to identify the signals that contain periodicity, and combine the output of these signals to produce a more robust frequency estimation.

By forming a general framework for periodicity detection in images and video, other application-specific problems can be converted to a similar format as the problems outlined in this chapter. For example, the method of detecting ties that was presented in Section 2.4 also utilizes a one-dimensional signal that is created by aggregating the results of two-dimensional Gabor filters. It is application-specific, but the method presented is inspired by the general framework of the more fundamental problems presented in this chapter.

This chapter provides: (1) a problem definition that summarizes image and video periodicity detection, (2) a general framework for obtaining one-dimensional signals from 2D images and 2-D+time videos, (3) a framework for combining those signals into one estimate of periodicity, and (4) a method for segmentation of the periodic object in images, and any object producing a periodic activity in video. There are many parallels between the methods for detecting periodic objects in images, detecting translating objects in video, and detecting objects producing a periodic activity in video.

6.1 Motivation

We outline the three problems involving periodicity detection in images and video, and propose a solution using spectral estimation techniques.

We summarize the three problems to solve as follows:

- Images: When an image is periodic in spatial locations, we must find the direction(s) of periodically repeating components, and segment all occurrences of the components that produce this periodicity.
- Video (translational): A translating moving object produces a phase modulation in the spatial frequency domain. Spatial frequency signals evolve periodically in time, but with a constant periodicity (thus, instantaneous frequency is constant at all times). We must detect the

translating velocities by detecting the periodicity of the spatial frequency signals as they evolve over time, and then segment the objects that produce the velocities corresponding to the detected values.

- Video (periodic): An object that is moving periodically will produce spatial frequency phase modulation that is periodic. That is, when the instantaneous frequency is examined for each spatial frequency, it will be periodic. We must detect which spatial frequencies exhibit periodic instantaneous frequency, compute the period of those instantaneous frequencies, and segment the objects that produce this.

The images and video are modeled as follows:

6.1.1 Images

A model for the image, with periodically repeating components, is created. A two-dimensional image $I(\vec{x})$, where $\vec{x} = (x, y)$, consists of both periodically repeating components, C_k, and non-periodic parts of the image, $\eta(\vec{x})$. Each component, C_k, is described as $C_k = \{\mathbf{p_k}, T_k, \upsilon_k\}$, where $\mathbf{p_k}$ is the two-dimensional appearance, T_k is the periodicity, and υ_k is the phase offset. Periodicity is defined as the number of pixels between consecutive component occurrences, as measured from a fixed point in that component. Phase offset indicates the location of the first instance of a repeating component with respect to the left-most pixel at $x = 0$.

The two-dimensional image, $I(\vec{x})$, is described as

$$I(\vec{x}) = \sum_{k=1}^{K} \sum_{z \in \vec{Z}_k} \delta(\vec{x} - (\vec{x}_k + z\vec{T}_k + \vec{\upsilon}_k))\mathbf{p}_k(\vec{x}_k) + \eta(\vec{x}) \tag{6.1}$$

where $\vec{Z}_k$ represents the two-dimensional range of z for which the component instance, k, is contained within the image $I(\vec{x})$, and $\eta(\vec{x})$ is the part of the image row that does not repeat. The term $\vec{x}_k$ refers to each component's internal coordinate system, which has the equivalent size as $\mathbf{p}_k(\vec{x}_k)$. We assume that the image itself is composed primarily of the K repeating components, $\mathbf{C}$, where $C_k = \{\mathbf{p_k}, T_k, \upsilon_k\}$.

6.1.2 Video

Recent work has shown that spatial localization can be computed if motion adheres to a rigid model of moving objects in a video [29]. We obtain the spatial Fourier transform of each video frame at time t, $I_t(\vec{\omega})$, and model it as

$$I_t(\vec{\omega}) = \sum_{l=1}^{L} O_l(\vec{\omega}) e^{j\vec{\omega}^T \vec{\rho}_l(t)} + V_{noise}(\vec{\omega}) - V_{back}(\vec{\omega}) \tag{6.2}$$

where there are L objects, each with spatial Fourier transform $O_l(\vec{\omega})$, and each with displacement $\vec{\rho}_l(t)$ (with respect to its position at $t = 0$). $V_{back}(\vec{\omega})$ is occluded background and $V_{noise}(\vec{\omega})$ is noise. Each frame is $N_x \times N_y$ pixels. This was described in [29]. Equation 6.2 models rigid objects, since $O_l(\vec{\omega})$ is time invariant.

6.1.3 Periodic Activity in Video

A periodic activity is characterized by the rigid objects $O_l(\vec{\omega})$ (from Equation 5.1) translating with a periodic displacement $\rho_l(t)$. This periodic displacement results in the an instantaneous frequency in Equation 5.1 that is periodic. We model this instantaneous frequency as

$$\vec{\rho}_l(t) = E\{\vec{\rho}_l(t)\} + \vec{a}_l cos(\vec{f}_l t + \vec{\psi}_l) \tag{6.3}$$

where $E\{\vec{\rho}_l(t)\}$ represents the expected or mean value of the displacement for object l, $\vec{f}_l$ is the frequency of the periodic signal, $\vec{a}_l$ is the amplitude, and ψ_l is the offset. The expected value, $E\{\vec{\rho}_l(t)\}$, is simply the translational motion of the object, while $\vec{a}_l$ is the expected amount of displacement due to the periodic motion.

6.2 Obtaining One-Dimensional Estimates

One-dimensional estimates for periodicity are obtained from the 2D images and 2D+t videos. An efficient mapping that captures the property we are looking for, namely a signal that oscillates at the same frequency as the repeating object or activity, is described for both images and video.

6.2.1 Images

As shown in Chapter 3, the one-dimensional estimates are composed of the projections along θ from various Gabor filters defined by ϕ and B. That is, the one-dimensional signal, $s_{\phi,B,\theta}(\lambda)$, is defined as

$$s_{\phi,B,\theta}(\lambda) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} G_{\phi,B}(x,y)\delta(\lambda - x\cos\theta - y\sin\theta)\,dx\,dy \tag{6.4}$$

If the one-dimensional signal $s_{\phi,B,\theta}(\lambda)$ is periodic, then we conclude that some object is repeating. Since the Gabor-filtered image, $G_{\phi,B}(x,y)$, is used instead of the original image $I(x,y)$, the spatial periodicity of certain textures and gradients will be detected, even if the color and illumination is ambiguous.

6.2.2 Video (Translational)

The object o_l is assumed to be constant in time, and all o_l are defined at the initial time t_0. Thus, at time t_0, Equation 5.1 becomes

$$I_{t_0}(\vec{\omega}) = \sum_{l=1}^{L} O_l(\vec{\omega}) + V_{noise}(\vec{\omega}) - V_{back}(\vec{\omega}) \tag{6.5}$$

At time t_0, we assume that the object has displaced by zero, that is, $\vec{\rho_l}(t_0) = 0$. At all subsequent times, each displacement $\vec{\rho_l}(t)$ results in a phase shift $O_l(\vec{\omega})e^{j\vec{\omega}\vec{\rho_l}(t)}$.

Displacement can also be expressed as velocity, where $\vec{\rho_l}(t) - \vec{\rho_l}(t_0) = \int_{t_0}^{t} \vec{\nu_l}(t)$. If velocity is constant, then we represent this constant velocity as $v_l(t) = \nu_l \;\; \forall t$. Thus, $\vec{\rho_l}(t) = (t - t_0)\vec{\nu_l}$. Letting $t_0 = 0$, a translational object l has its displacement computed as

$$\vec{\rho_l}(t) = t\vec{\nu_l} \tag{6.6}$$

For an object traveling at a constant velocity, an individual given frequency, $\vec{k}$, produces a frequency response

$$\mathcal{F}_{\vec{k}}(i_t(\vec{x})) = I_t(\vec{\omega_k}) = \sum_{l=1}^{L} O_l(\vec{\omega_k}) e^{j\vec{\omega}_k^T \vec{\nu}_l t} + V_{noise}(\vec{\omega}) - V_{back}(\vec{\omega}) \tag{6.7}$$

To ease the computation, we choose a particular frequency where only one component, either the x or y component, has a non-DC value. That is, each frequency is $\vec{\omega_k} = (0, \mu_{k,y})$ for some k, or $\vec{\omega_k} = (\mu_{k,x}, 0)$ for some k. The resulting signal for $\vec{\omega_k} = (\mu_{k,x}, 0)$ is

$$O_l(\mu_{k,x}, 0)e^{j\mu_{k,x}\nu_{l,x}t} \tag{6.8}$$

A complex signal $e^{j2\pi f_0 t}$ has a frequency of f_0 and a period of $T_0 = \frac{1}{f_0}$. In the discrete fourier transform (DFT), a spatial frequency $(\mu_{k,x}, 0)$ corresponds to a spatial frequency bin $(k_x, 0)$. Equation 6.8 can be rewritten as

$$O_l(\mu_{k,x}, 0)e^{j2\pi \frac{k_x}{N_x}\nu_{l,x}t} \tag{6.9}$$

where N_x is the width of each frame and $\nu_{l,x}$ is the velocity of the l-th object in the x-direction.

The signal $O_l(\mu_{k,x}, 0)e^{j2\pi \frac{k_x}{N_x}\nu_{l,x}t}$ oscillates with frequency $f_{l,x} = \frac{k_x V_{l,x}}{N_x}$, and period $T_{l,x} = \frac{N_x}{k_x V_{l,x}}$.

One can therefore apply spectral estimation to the spatial frequencies of $\mathcal{F}_k\{I_t(\vec{x})\}$ to estimate the constant translational velocity of the objects o_l. This method will become cumbersome for many objects L. The phase correlation method shown in Section 2.3 is preferable for detecting interframe velocities.

Spectral estimation is used in a synthetic example, shown in Figure 6.1, to detect constant translational motion. All spatial frequencies $\vec{k}_x = (\mu_x, 0)$ and $\vec{k}_y = (0, \mu_y)$ are examined. The phase component of Equation 6.9, $j2\pi\frac{k_x}{N_x}\vec{\nu}_{l,x}t$, has a frequency $\frac{k_x\vec{\nu}_{l,x}}{N_x}$ and period $T = \frac{N_x}{k_x\vec{\nu}_{l,x}}$ that will be detected by spectral estimation. The measure of correctness from Equation 4.3.1 is utilized again to confirm that the signal oscillates accordingly.

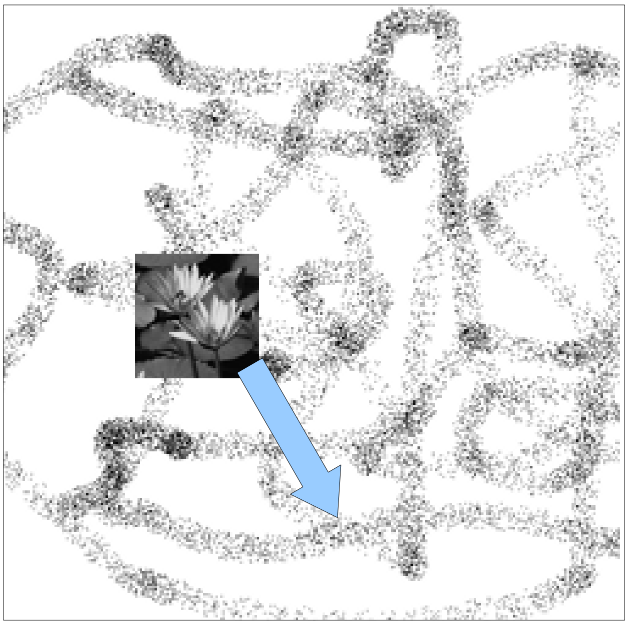


Figure 6.1: Frame from synthetic video. Arrow indicates direction.

6.2.3 Video (Periodic)

When an object o_l moves periodically, the velocity $\vec{\nu}_l(t)$ will no longer be constant as in the case of translational motion. Instead, it will itself be a periodic function. For the lth object's motion in the x direction, the phase of each of spatial frequency in the x-direction is now

$$\rho_{l,x}(t) = a_l cos(f_{l,x}t + \psi_{l,x}) \tag{6.10}$$

The instantaneous frequency of Equation 6.10 can now be examined for evidence of a periodic function. An STFT is used. Since the output of this is a two-dimensional function, we utilize the one-dimensional spectral centroid projection from Chapter 5.

The periodically moving object can itself be translating. This is easy to detect in the instantaneous frequency, as the resulting signal will have a non-zero mean value.

A measure of correctness is applied to confirm that the temporal signal is in fact phase-modulated by a periodic signal. The measure of correctness from Equation 4.3.1 can be used to measure this.

6.3 Aggregating the Frequency Estimates

The frequency estimates obtained in Section 6.2 can either be detected separately and subsequently grouped, or they can be computed concurrently by operating on a signal that is formed by superposing several one-dimensional signals. This section contains the methods used to aggregate the periodicity estimate in images and video.

6.3.1 Images

The individual estimates for $s_{\phi,B,\theta}(\lambda)$ are aggregated into one function, $\Lambda(\phi,\theta)$, by computing a quality metric $Q_{\phi,B,\theta}$ for all (ϕ,B,θ) and summing across all block sizes B. The values $(\phi_0,\theta_0) = \arg\max_{\phi,\theta}\Lambda(\phi,\theta)$ are chosen. The frequency is then estimated from a reconstructed image, in the direction of θ_0, where the reconstruction occurs using the ϕ_0 that exhibited periodicity. The final estimate of T will be formed from a filtered image in the direction ϕ_0, projected in the direction θ_0, with a B that resulted in $B_0 = \arg\max_B Q_{\phi_0,B,\theta_0}$.

Our method involved decomposing images into Gabor-filtered images, projecting and summing these images in the direction θ, and estimating the periodicity from that. This was motivated by the Fourier slice theorem, which states that the one-dimensional Fourier transform of the projection of an image along the direction θ is equivalent to the *Fourier slice* of that image in the Fourier domain at angle θ. Although we applied the MUSIC algorithm to that one-dimensional projection, this was only to avoid the spectral spread that the Fourier domain would produce. MUSIC is used because there is an assumption of a small number of periodic signals. In the spatial Fourier domain, the edges and other objects that are occurring in the θ direction can be isolated by decomposing the Fourier domain at angle θ. This is illustrated in Figure 6.2(a), where it is indicated that along the spatial Fourier direction θ, there should be dominant peaks. These correspond to a repeating object's frequency, and the θ where this occurs corresponds to the direction of repetition. Since we first filtered the image into many Gabor-filtered images, the stacked effect of the image indicates that several of these estimates exist.

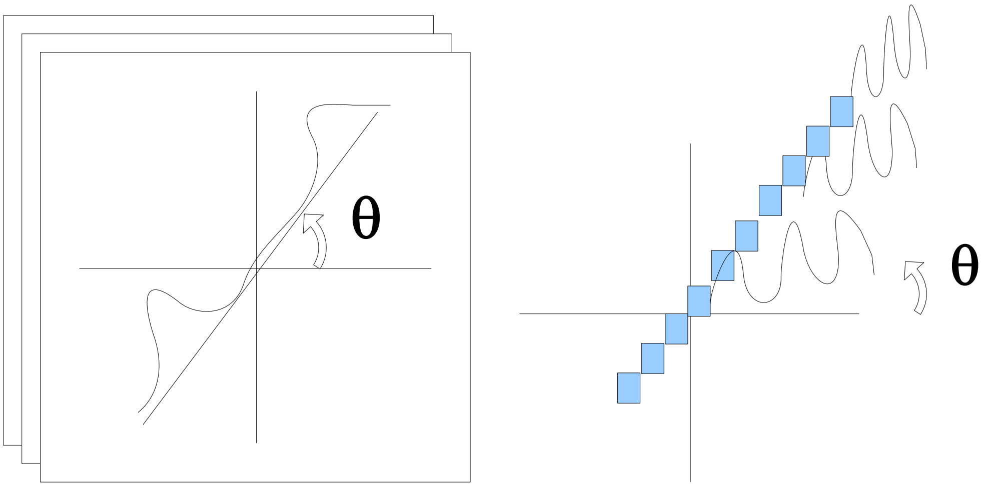


Figure 6.2: Relationship between direction of periodicity (θ) and the spatial Fourier transform. (a) In images, a periodic object will result in high energy spatial frequency along the orientation of periodicity, and (b) in video, a translating object will result in periodic frequencies along an orientation θ, where all periodicities will be related by a constant value.

6.3.2 Videos (Translational)

The estimates of periodicity are obtained from the individual results of the spatial frequency bins. T is estimated from several spatial frequency bins, and the most commonly obtained T is determined. Extending the analysis of Section 6.2.2, the time-induced frequency of the spatial frequency $(\mu_{k,x}, \mu_{k,y})$ is $\frac{k_x \nu_{l,x}}{N_x} + \frac{k_y \nu_{l,y}}{N_y}$. If the two spatial frequencies are related by some constant, say that $\mu_{k,y} = m\mu_{k,x}$. Then the modulation term becomes

$$\begin{aligned} \frac{k_x \nu_{l,x}}{N_x} + \frac{k_y \nu_{l,y}}{N_y} &= \frac{k_x \nu_{l,x}}{N_x} + \frac{(mk_x)\nu_{l,y}}{N_y} \\ &= k_x \left(\frac{\nu_{l,x}}{N_x} + \frac{m\nu_{l,y}}{N_y} \right) \qquad (6.11) \end{aligned}$$

Thus, if the spatial frequencies are related by $\mu_{k,y} = m\mu_{k,x}$, then their periodicity that is induced by $\nu_{l,x}$ and $\nu_{l,y}$ will be related by a constant value k_x. For $|\theta| \le \frac{\pi}{4}$, $m = \tan\theta$. An example of spatial frequency bins that are related in this manner is shown in Figure 6.2(b).

If a video contains translational motion, then the spatial frequencies along θ should all have time-related periodicities that are proportional to each other. Thus, each spatial frequency can be used as an estimate. For compu-

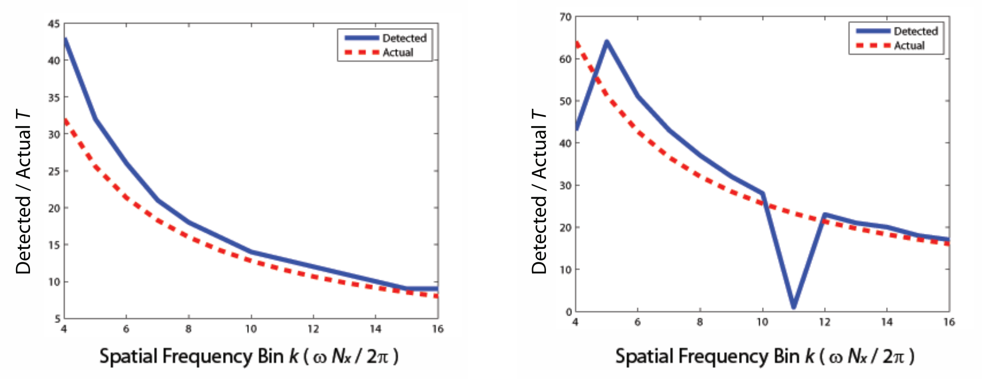


Figure 6.3: Frequency detection for horizontal and vertical motions in synthetic video.

tational ease, the estimations used in the experiment for Figures 6.3(a) and (b) are acquired at $\theta = 0$ and $\theta = \frac{\pi}{2}$ respectively.

6.3.3 Videos (Periodic)

When spatial frequencies are related by $\mu_{k,y} = m\mu_{k,x}$, the time-induced periodicity among them is related by a constant, as stated in Section 6.3.2. Thus, the effect on instantaneous frequency is as follows. From Section 6.2.3, $\rho_{l,x}(t) = a_l cos(f_{l,x}t + \psi_{l,x})$ and $\rho_{l,y}(t) = a_l cos(f_{l,y}t + \psi_{l,y})$.

The result of a MUSIC algorithm that is applied to this will be the detection of the frequencies $\vec{f}_{l,x}$ and $\vec{f}_{l,y}$. However, all frequencies that are related by $\mu_{k,y} = m\mu_{k,x}$ will have the same instantaneous frequency modulations $f_{l,x}$ and $f_{l,y}$. Thus, we should utilize all frequency bins along some θ to form an estimate of $f_{l,x}$ and $f_{l,y}$.

As explored in Chapter 3, from a received signal $\mathbf{y}_d$, the Toeplitz matrix $\mathbf{Y}_d$ can be created. Then, $\mathbf{Y}$ is created, where $\mathbf{Y} = \sum_d \mathbf{Y}_d$. An autocorrelation matrix is then formed, $\mathbf{R}_y = \mathbf{Y}\mathbf{Y}^H$, and the MUSIC algorithm is subsequently applied. To estimate the values of $\vec{f}_{l,x}$ and $\vec{f}_{l,y}$ for a given direction θ, this method is applied to several frequency bins $d = (\mu_1, \mu_2)$ along the angle θ.

Let $z_{(\mu_1,\mu_2)}(t)$ be the spatial frequency (μ_1, μ_2) as it evolves in time, that is,

$$\begin{aligned} z_{\mu_1,\mu_2}(t) &= \sum_{l=1}^{L} O_l(\mu_1,\mu_2) e^{j(\mu_1 t v_{l,x}(t)+\mu_2 t v_{l,y}(t))} \\ &+ V_{noise}(\mu_1,\mu_2) - V_{back}(\mu_1,\mu_2) \end{aligned} \tag{6.12}$$

Applying an STFT yields

$$|Z_{\mu_1,\mu_2}(p,t)| = |\sum_{h=0}^{H-1} z_{\mu_1,\mu_2}(t+h)w(h)e^{-j\omega_p h}| \tag{6.13}$$

We compute the spectral centroid, as we did in Chapter 5, for the frequency (μ_1, μ_2):

$$C_{\mu_1,\mu_2}(t) = \sum_{p=1}^{M} \frac{p|Z_{\vec{\mu}}(p,t)|}{\sum_{p=1}^{M} |Z_{\vec{\mu}}(p,t)|} \tag{6.14}$$

This will serve as our input signal to MUSIC, as in Chapter 3. We again form a Toeplitz matrix from our signal, $y = C_{\mu_1,\mu_2}(t)$, and denote it as $\mathbf{Y}_{k_{\mu_x},k_{\mu_y}}$. These matrices are then added for all k_{μ_x}, k_{μ_y} that lie along an angle θ, and the matrices are accumulated as

$$\mathbf{Y} = \sum_{k_{\mu_y}/k_{\mu_x}=\tan\theta} \mathbf{Y}_{k_{\mu_x},k_{\mu_y}} \tag{6.15}$$

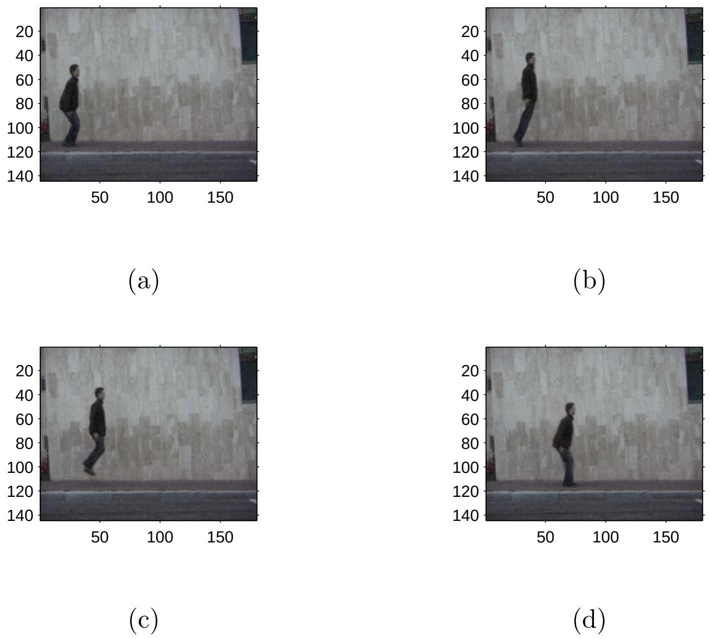
From this, we can find the MUSIC algorithm according to Chapter 3. The autocorrelation matrix is formed

$$\mathbf{R_y} = E\{\mathbf{y}\mathbf{y}^H\} = \mathbf{Y}\mathbf{Y}^H = \mathbf{A}\mathbf{R_s}\mathbf{A}^H + \sigma^2\mathbf{I} \tag{6.16}$$

and after performing the MUSIC algorithm, an estimate for periodicity as determined by all frequencies along line θ, T_θ, is given.

For example, for the Weizmann video sequence of *eli-jump.avi*, various orientations of θ are examined using spectral estimation to determine which ones exhibit periodicity. As shown in Figure 6.4, there is periodicity in a person jumping, which repeats approximately every 12 frames of the video.

The video is now examined at several orientations θ. The periodicity of $T \approx 14$ is detected more prominently at certain θ, as indicated by the various SNR responses in Figures 6.5(a)-(c).



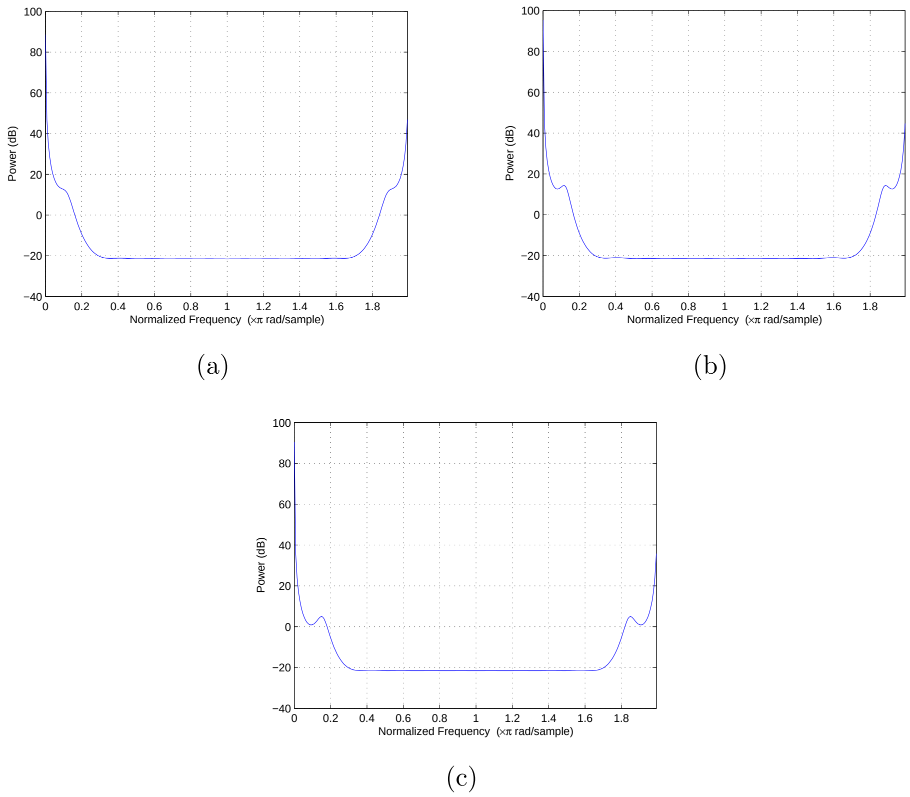
(a)

(b)

(c)

(d)

Figure 6.4: (a) Frame 1 (b) Frame 5 (c) Frame 9 (d) Frame 15.



(a)

(b)

(c)

Figure 6.5: (a) $\theta = 0.3927$, (b) $\theta = 1.5708$, (c) $\theta = 2.3562$.

The SNR alone cannot completely determine which orientation θ is producing a periodic signal. To determine this, T_{μ_x,μ_y} is computed for (μ_x, μ_y) at many orientations. The median value is computed, T_{MV}. The set of consecutive orientations that produce a periodicity $T_\theta \approx T_{MV}$ is computed, and the longest range of consecutive θ is determined and stored in a vector $\mathbf{\Theta}$. These $\mathbf{\Theta}$ represent the spatial frequencies that produce a periodic signal. These are going to originate from rigid objects that maintain a strong edge oriented at a θ contained within $\mathbf{\Theta}$.

6.4 Segmentation

The estimate of periodicity for both images and video utilizes the frequency domain, so the estimated periodicity, T, will not contain any spatial localization information. For this reason, once the global properties of expected T are known, the segmentation of the object presents a challenge.

6.4.1 Images

In Chapter 3, two methods for segmentation were proposed. A gradient-based method was suitable for objects that were defined primarily by their edges, and a block-based method was suitable for larger objects.

6.4.2 Video (Translational)

In [29], a method was introduced that involved a least-squares solution for object reconstruction, followed by a block-based method for segmentation. For video frames that are $N_x \times N_y$ in size, the result of this least-squares formulation is also $N_x \times N_y$ in size; however, the resulting frames contain a spatial-frequency filtered version of the image where each reconstructed object has sharp gradients, and thus appears more visible.

The segmentation in [29] is implemented in a block-wise manner. Cross-correlation is used to determine which image patches of the reconstructed objects are most similar to the original image. These image patches are then labeled as the object.

A gradient-based method could also be implemented in a similar fashion to Chapter 3. Gradient-based methods work well because the reconstructed object, having been formed in the spatial frequency domain, will have sharp edges where the spatial-frequency domain was successfully reconstructed. The gradient-based method used for segmenting periodic activities [24] is also applicable to translating objects in video.

6.4.3 Video (Periodic)

The segmentation procedure for video containing periodic activity is identical to what was presented for translational video in Section 6.4.2. Additionally, we can isolate the θ that produced the dominant periodicity rather than the dominant velocity.

One can isolate the θ with spatial frequencies which predict the same T. These correspond to edge orientations that had the most evidence of periodicity. The gradient method works well if there is consistent periodicity detection along an angle θ (therefore strong, oriented gradients). An example of the most dominant θ from the Weizmann movie *daria-skip.avi* is shown in Figure 6.6(a), and the original video frame is shown in Figure 6.6(b).

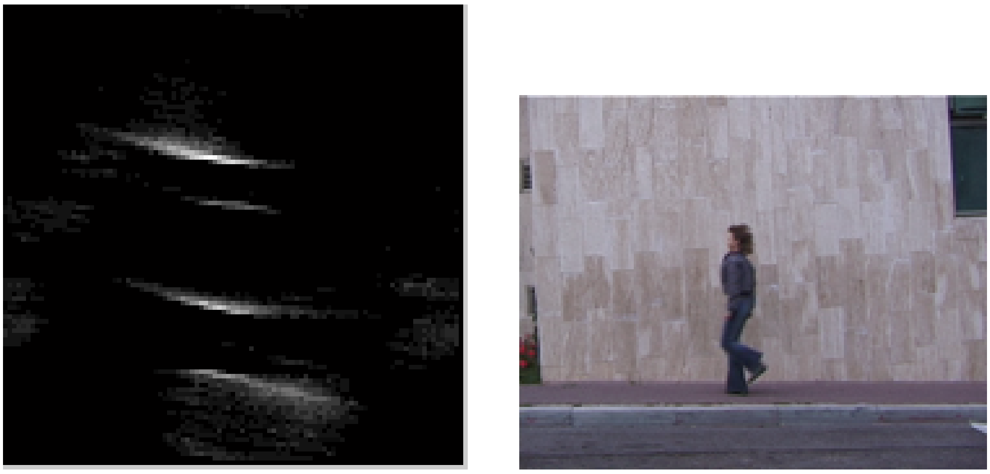


Figure 6.6: (a) Reconstruction of the strongest orientations that are rigid and periodic, and (b) video frame from original video.

6.5 Conclusion

In this chapter, we produced a formulation for detecting periodicity in images and video which utilizes spectral estimation. We demonstrated how one-dimensional projection signals can be created from images and video. Using the Fourier slice theorem as motivation, one can gain valuable insight from spatial frequencies that lie along a line at angle θ in the spatial Fourier domain. There are three problems which we identified as similar problems with similar solutions: periodic object detection in images, translational moving object detection in video, and periodic moving object detection in video. Many other applications can utilize similar solutions, particularly applications where the inspected image or video can be summarized as a one-dimensional signal.

CHAPTER 7

SEGMENTATION-BASED PERCEPTUAL IMAGE QUALITY ASSESSMENT

Computational representation of perceived image quality is a fundamental problem in computer vision and image processing, which has assumed increased importance with the growing role of images and video in human-computer interaction (HCI) [8]. A system in which an algorithm can alert a railroad track inspector of a defect that he or she is likely to have overlooked is an example of HCI. This is a motivating reason to study the human visual system (HVS), and to analyze how the HVS perceives distortion.

It is well-known that the commonly used peak signal-to-noise ratio (PSNR), although analysis-friendly, falls far short of this need. We propose a perceptual image quality measure (IQM) in terms of an image's region structure. Given a reference image and its "distorted" version, we propose a "full-reference" IQM, called Segmentation-based Perceptual Image Quality Assessment (SPIQA), which quantifies this quality reduction, while minimizing the disparity between human judgment and automated prediction of image quality. One novel feature of SPIQA is that it enables the use of inter- and intra-region attributes in a way that closely resembles how the HVS perceives distortion. Experimental results over a number of images and distortion types demonstrate SPIQA's performance benefits.

7.1 Introduction

An IQM that creates a computational representation of perceived image quality is needed in computer vision and image processing, and has assumed increased importance with the growing role of images and video in human-computer interaction. PSNR is the de-facto standard for quality assessment due to its computational simplicity; however, its use of a pixel-based distance metric fails to capture the human-perceived qualities of image distortion.

Several other IQMs have been proposed recently [33], but none form a computationally robust method that mimics the HVS. Humans comprehend the contents of an image, and using mid-level techniques, an IQM should emulate the HVS by examining the structure of an image and quantifying distortion in terms of the perturbations to image structure. Segmentation is a mid-level vision technique that captures the structure of an image and achieves dimensionality reduction by dividing an image into regions that are defined by their shape, color, size, and texture. SPIQA, our proposed IQM, achieves superior results by using segment-based regions to quantify the distortion of an image in terms of image structure.

We compare SPIQA against PSNR and the three IQMs which had the best experimental performance in the recent survey paper [33]. Our IQM, like all three of the IQMs in [33], had superior results to PSNR. SPIQA not only outperformed the three IQMs in [33], but was able to train on only 13% of the database that was used in [33] and achieved lower root mean square error (RMSE) with respect to human opinion scores, even before the nonlinear regression fitting that was necessary in [33].

The rest of this chapter is organized as follows: Section 7.2 summarizes the previous work on IQM's, Section 7.3 gives a description of the underlying components that formulate the SPIQA measure, Section 7.4 presents experimental results of the algorithm, demonstrating its performance as compared to the IQMs that achieve the best results in [33], and Section 7.5 provides concluding comments and future goals.

7.2 Background

The formulation of IQMs for image quality assessment (QA) is an old field. IQMs can be divided into two categories, subjective or objective IQMs, according to the amount and form of human intervention involved. (1) A subjective IQM requires direct human intervention, since it is based on the cumulative judgment of a group of human observers. This type of IQM is heavily correlated with the observers' preferences.

(2) On the other hand, an objective IQM analyzes a distorted image and possibly a reference image, in the absence of any direct human intervention. Most IQMs are of this type. Objective IQMs are categorized as either "full-

reference" or "blind-reference" according to the availability of the reference image. The former category renders a quantitative measurement of the quality of an image in the absence of the reference. Most IQMs fall into the second category, where the measure is computed by comparing the reference and distorted images. In this paper, we will discuss a novel, "full-reference," region-based IQM, SPIQA.

Traditional objective IQM methods rely primarily on modeling and approximating the functionality of HVS in terms of well-known image processing operations. One of the first notable IQMs was the Just Noticeable Difference (JND) measure, which was developed in the seminal work by Lubin ([34]). JND and other traditional IQMs quantify the threshold of distortion that must be exceeded before a human can perceptually detect that this distortion has been imposed on the reference image. These methods tend to fall short of efficiently approximating the complex, nonlinear functionality of the HVS. Also, some methods of this IQM type rely on parameters that are dependent on experimental settings.

More recent objective IQMs are considered signal fidelity IQMs, since they compute the measure based on inherent features of the pair of images only, thus avoiding dependence on the experimental setup. In this work, we consider three IQMs: (a) the simplest and the de facto standard measure of peak signal-to-noise ratio (PSNR), and two signal fidelity IQM's that showed the best experimental performance in a recent survey [33]: (b) Multi-Scale Structural Similarity (MSSSIM) [35] and (c) Visual Information Fidelity (VIF) [36].

(a) PSNR uses a pixel-based distance noise. However, this method fails to capture the structure of distortions. Such structure plays an important role in perception of distortion by humans, and occurs in most applications (e.g. blocking artifacts due to JPEG compression).

(b) MSSSIM divides an image into rectangular blocks, or patches, and computes first and second order statistics for each patch. These results are combined to form the MSSSIM. It suffers from the following limitations. First, its use of first and second order moments does not suffice to represent the luminance distribution of image patches. Second, its "structural factor" is independent of the spatial distributions of image values. The spatial relationships within the image are not explored, which are known to be important to the HVS. This is evident from the phenomenon called *visual masking* [37], where respective luminance distributions and spatial localization of the image regions are able to mask or enhance the presence of a distortion in a specific region.

(c) VIF takes an information fidelity approach to the problem of image QA using wavelet decomposition. No explicit interaction between wavelet subbands is modeled. Instead, the subband-specific VIF measures are pooled together independently to render the final overall IQM. This independence counteracts a highly regarded factor in HVS literature ([37]), the *contrast sensitivity function* (CSF), which renders certain wavelet subbands (especially the lower and higher frequencies) less effective than others in determining image quality.

In this paper, we propose a signal fidelity IQM (SPIQA), which manipulates some of the important characteristics of the HVS, while remaining independent of subjective factors related to the experimental setup. We make use of the well-known psychophysical observation that human vision tends to concentrate on coherent image regions instead of arbitrary image blocks. Consequently, we propose to use features inherent to regions resulting from image segmentation. We impose inter- and intra-segment measures that take into account possible regional interactions that have been ignored by previous IQMs.

7.3 Region Based Image Quality Assessment

In this section, we describe how SPIQA is formulated. The major motivation for our measure is to incorporate image segments in its definition, which makes the quality measure depend on spatial structure in addition to image intensity values.

Image segmentation partitions an image into disjoint regions that contain pixels that are “similar” to each other, but “different” from the pixels of another region. The problem of efficient and perceptually correct segmentation is still an unsolved problem in computer vision, but there are numerous algorithms in the literature that approximate the segmentation. We use the segmentation algorithm implemented in [38]. In essence, this multi-scale segmentation algorithm proposes a region model characterized by a homogeneous region surrounded by ramp discontinuities. Thus, each segment at every photometric scale includes ramp and non-ramp pixels. In our implementation, we require that segmentation be performed at a single photometric scale and only on the reference image, where the same segment boundaries are also used for the distorted image.

Assuming that image segmentation is given, we compute the overall measure as a weighted sum of the regional image quality measures (RIQMs). These weights summarize the importance of each segment in determining the overall image quality. Before proceeding with in-depth explanation of SPIQA, we present an outline of its structure:

$$\text{SPIQA} = \sum_{\mathbf{seg}_i \subset \mathbf{I}_{ref}} \kappa_i \ \text{RIQM}_i \tag{7.1}$$

$$\kappa_i = \beta_1 \text{sal}_i + (1-\beta_1)\text{size}_i \ ; \quad \beta_1 \in [0,1] \tag{7.2}$$

where κ_i weighs the contribution of the RIQM in the i^{th} segment. It summarizes all inter-segment interactions by quantifying the importance of the corresponding segment in terms of its overall saliency and size. RIQM$_i$ is the regional quality measure and it summarizes all intra-segment interactions according to how the HVS perceives the quality of each segment independently.

7.3.1 Inter-Segment Interactions: κ_i

The term κ_i is expressed as a linear combination of the following two normalized factors:

$$\text{size}_i = \frac{\#\text{ of pixels in seg}_i}{\#\text{ of pixels in }\mathbf{I}_{ref}}, \ \text{sal}_i = \frac{\text{saliency of seg}_i}{\text{saliency of }\mathbf{I}_{ref}}$$

Here, saliency is computed in accordance to human visual attention as described in [39] (refer to Figure 7.1). We justify the use of saliency from a human perception point of view. Humans concentrate on high-level features of an image to identify its contents; however, the saliency algorithm computes the saliency map using low-level features and on a pixel basis. By incorporating the pixel-based saliency map into our coherent regions, we are able to incorporate higher-level features that better represent the focus of human attention. By virtue of κ_i's, the effects of distortion on the image quality are more influenced by the distortions in the most salient regions. To the best of our knowledge, the use of such segment interaction is novel to non-traditional IQM formulation, which usually assumes independence between neighboring blocks or bands.

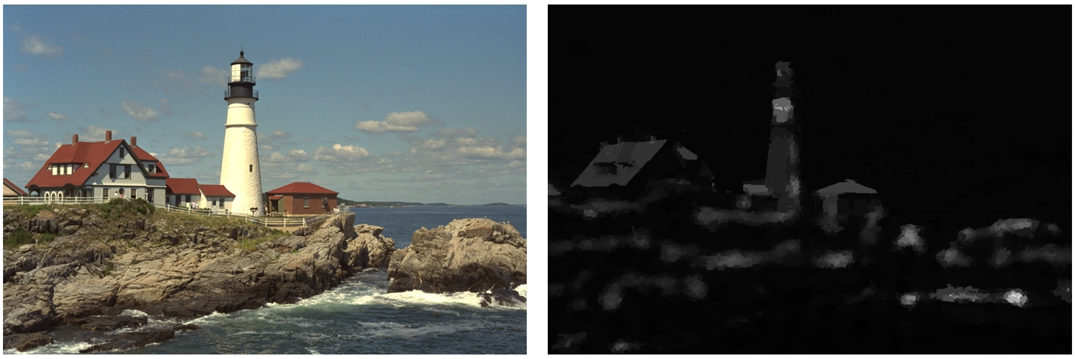


Figure 7.1: The reference image is on the left and its corresponding regional saliency map is on the right. Lighter regions indicate higher saliency.

7.3.2 Intra-Segment Interactions: RIQM$_i$

This section highlights the intra-segment interactions, which capture the "similarity" between the reference and distorted segment. The YCrCb color space (primarily the luminance component) is used, since it best approximates the HVS color perception among common color spaces. RIQM is defined as the product of three factors: RIQM$_i$ $= (\Delta\Gamma_i)^{\beta_3}(\Delta H_i)^{\beta_4}(\Delta NM_i)^{\beta_5}$. These factors quantify the similarity in histogram, mutual information, and spatial variation between the reference and distorted segment. They are properly normalized to take on values in the range [0,1].

Gradient Similarity ($\boldsymbol{\Delta\Gamma}$): We differentiate the reference and distorted segment in terms of difference in Sobel gradient energy. This structural term is absent in PSNR, MSSSIM, and VIF. Based on [38], a segment contains either significant ramp or non-ramp pixels, which are distinguished according to the variations of their luminance values with their neighbors. In fact, it is important to evaluate the perceptual effects of distortion on these two kinds of pixels. Therefore, we can write $\boldsymbol{\Delta\Gamma}$ of a segment (p) as a linear combination of two terms: the gradient similarity at significant ramp pixels ($\Delta\mathbf{\Gamma_{p_r}}$) and at non-significant ramps ($\Delta\mathbf{\Gamma_{p_{nr}}}$) as in (7.3) with $\beta_2 \in [0, 1]$.

$$\Delta\Gamma_p = \beta_2 \Delta\Gamma_{p_r} + (1 - \beta_2)\Delta\Gamma_{p_{nr}} \tag{7.3}$$

where $\Delta\mathbf{\Gamma_{p_r}}$ and $\Delta\mathbf{\Gamma_{p_{nr}}}$ are defined as

$$\Delta\Gamma_{p_r} = \frac{2\overrightarrow{\Gamma}_{\text{ref}_{p_r}} \cdot \overrightarrow{\Gamma}_{\text{dis}_{p_r}}}{\|\overrightarrow{\Gamma}_{\text{ref}_{p_r}}\|^2 + \|\overrightarrow{\Gamma}_{\text{dis}_{p_r}}\|^2 + \epsilon} \tag{7.4}$$

$$\Delta\Gamma_{p_{nr}} = \frac{2\overrightarrow{\Gamma}_{\text{ref}_{p_{nr}}} \cdot \overrightarrow{\Gamma}_{\text{dis}_{p_{nr}}}}{\|\overrightarrow{\Gamma}_{\text{ref}_{p_{nr}}}\|^2 + \|\overrightarrow{\Gamma}_{\text{dis}_{p_{nr}}}\|^2 + \epsilon} \tag{7.5}$$

The Sobel gradient energy, Γ, that is used in (7.3)-(7.5) is computed from the gradient of the image I: $\Gamma_x(i) = \|\nabla I_x(i)\|$, where x defines the specific region components $(p_r|p_{nr})$ and the specific image type (dis|ref).

Histogram Similarity ($\Delta\mathbf{H}$): This term is a non-structural factor that measures the difference in the distribution (estimated by a histogram) of the luminance values of the pixels within the reference and distorted segment. We define it as $\Delta H_p = \frac{2\overrightarrow{H}_{\text{ref}_p} \cdot \overrightarrow{H}_{\text{dis}_p}}{\|\overrightarrow{H}_{\text{ref}_p}\|^2 + \|\overrightarrow{H}_{\text{dis}_p}\|^2 + \epsilon}$. This factor improves on the SSIM measure, since the difference in luminance distributions encompasses more information than simply the 1st and 2nd moments.

Normalized Mutual Information Similarity ($\Delta\mathbf{NM}$): This is another non-structural factor, which builds on the assumption made in [36] that the HVS reacts to the loss in mutual information between the reference and distorted images. It normalizes the segment mutual information by the entropy of the reference segment. We define it as $\Delta NM_p = \frac{I(I_{\text{ref}_p}; I_{\text{dis}_p})}{H(I_{\text{ref}_p})}$.

We determine all five β values by minimizing the squared error between the resultant SPIQA and the experimental human decisions, in difference mean opinion score (DMOS) format, over a set of N training image pairs as shown below.

$$\overrightarrow{\beta}^* = arg\ min \sum_{t=1}^{N} |\text{DMOS}(t) - \text{SPIQA}(t, \overrightarrow{\beta})|^2$$

Contributions: The contributions of our proposed IQM are threefold: (1) it uses image segmentation to delineate coherent regions of human attention, (2) it quantifies both inter- and intra-region interactions in a manner that conforms to certain functional aspects of the HVS, and (3) it quantifies the quality of an image segment via local (e.g. at ramp and non-ramp pixels) and global features that represent both the structure and the content of each segment.

7.4 Experimental Results

We evaluate our IQM on the LIVE database used by [33], which presents the most recent and comprehensive survey of the performance of various IQMs available in the literature. We will compare our proposed IQM with the ones examined in [33] (i.e. MSSSIM and VIF) using the experimental human results that [33] presents in normalized DMOS format.

First, we show empirical evidence that demonstrates the improvement in an IQM's performance when it is applied to image segments instead of rectangular image blocks. This justifies our claim that using segmented regions for QA produces a measure that is more correlated to the HVS than using the non-structural image areas described in prior work. For example, Table 7.1 shows the improvement in performance of a segment-based SSIM over the original SSIM, which is in part due to the fact that the deterioration due to block boundaries is significantly alleviated.

Table 7.1: Original SSIM vs. segment SSIM

	CC	RMSE
Original SSIM	0.7815	18.75
Segment SSIM	0.9257	6.120

Next, we learn the β values from 13% of the image pairs in the database (i.e. 20 from each of the five distortion types) and compare the performance of SPIQA against that of MSSSIM and VIF on the whole database. Figure 7.2 shows each raw quality measure before and after nonlinear regression (as described in [33]). For visual purposes, we only consider a portion of the LIVE database in these plots. The impact of nonlinear regression on both VIF and MSSSIM is quite significant, while it is incremental for SPIQA.

Table 7.2 summarizes the performance of SPIQA, VIF, MSSSIM, and PSNR. Here, all the database samples are used for training in order to compare to the experimental results reported in [33]. These experiments show that SPIQA outperforms other measures of image quality, despite the introduction of nonlinear regression. In fact, piecewise polynomial regression results in more significant overall performance for SPIQA. This improvement is primarily due to the similarity between the spread of the raw SPIQA measures and the DMOS values.

Table 7.2: RMSE comparison between SPIQA and (PSNR + MSSSIM + VIF)

	PSNR	MSSSIM	VIF	SPIQA
JPEG2000	10.61	5.999	5.093	5.076
JPEG	12.17	5.465	5.318	5.585
WN	4.669	6.358	4.360	3.920
GBlur	11.44	5.823	3.991	4.117
FF	12.97	10.40	6.855	3.519
All Data	13.43	9.369	8.246	6.546

Table 7.3: SPIQA weights from 100 samples

	β_1	β_2	β_3	β_4	β_5
All Types	0	0.872	2.407	0.670	0.255

Table 7.3 shows the numerical values for the estimated β values. They represent the relative importance of each individual value in determining image quality. From these results, we can make some remarks about the importance of each factor in the SPIQA measure from its corresponding β value. This dependence highlights not only the mechanism of the distortion, but also the location of its manifestation in the image (e.g. at object boundaries, “flat” intensity regions, or over the entire image). The β values and their meaning are described as follows:

- β_1: Regional saliency is the single inter-regional factor to be maintained. This term already has an inherent relationship with segment size, as it is the normalized sum of all saliency values within a segment.

- β_2: Significant ramp pixels tend to be more effective in detecting change in image quality especially for distortion types that impose structured alteration at locations close to strong edges (e.g. JPEG2000).

- β_3: $\Delta\Gamma$ plays the most influential role in QA. This is due to the fundamental impact of structured organization on human visual perception.

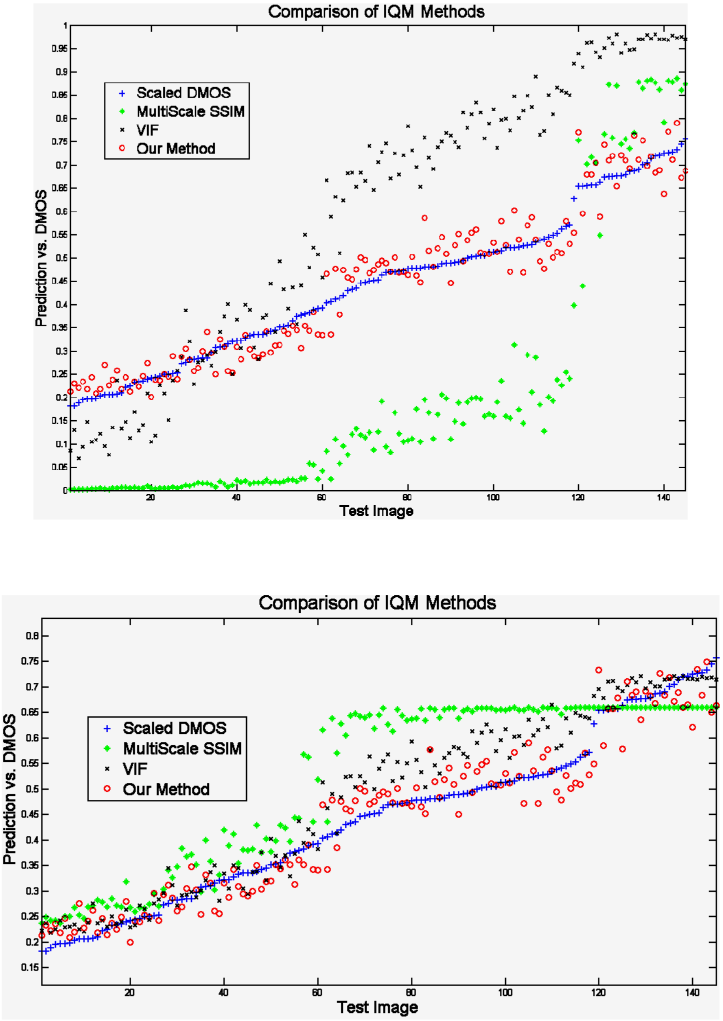


Figure 7.2: SPIQA, VIF, and MSSSIM, before (top) and after (bottom) regression.

- β_4: ΔH is critical in evaluating distortion types that produce significant disruptions in regional luminance distribution (e.g. Gaussian blur). But, the HVS seems to tolerate more change in ΔH than $\Delta\Gamma$.

- β_5: ΔNM is the least important factor, despite its informational description of human visual judgment.

7.5 Conclusion and Future Work

In this paper, we presented a novel segmentation based image quality measure, which models both inter- and intra-segment relationships, thus capturing the HVS characteristics more effectively than previous IQMs. SPIQA improves over the state-of-the-art quality measures by reducing the gap between automatic prediction and human judgment of image quality.

For future work, we would like to explore the performance of the HVS when performing certain inspection tasks, particularly for railroad track inspection. We would be interested in detecting not only distortion perception, but also object fixation as it affects inspection. If one were to quantify a per-region human attention measure, it could assist in developing a computer vision system whose primary goal is to alert an inspector to defects that he or she is likely to overlook.

CHAPTER 8

CONCLUSION

Computer vision shows great promise in railroad track inspection. Using computer vision algorithms, many defect detection and object identification solutions have been developed. We introduced an algorithm that detects and segments periodically occurring objects in images. Initially, this algorithm was developed for video data that involved objects that were oriented only along one direction. The periodic objects were detected in a row-wise manner, and spectral estimation was used. Spectral estimation is a valuable signal processing technique that allows us to extract periodic signals from a one-dimensional signal, and its robustness to noise allows it to effectively estimate periodicity in real-world inspection images.

An extension of the method for detecting periodicity along one dimension was presented that allows us to detect periodic objects that are oriented along any direction. Additionally, no prior knowledge about the object's size, shape, or appearance is required. One-dimensional projections were obtained from Gabor-filtered images, and orientation of the objects and direction of periodicity was computed.

We presented a method for periodic activity recognition that also utilizes the Fourier domain to form one-dimensional signals. From these one-dimensional signals, which are computed in the Fourier domain, one can detect periodicity and classify activity. Though periodic object detection in images and periodic activity recognition in video may seem different, the solutions are remarkably similar. The similarity was examined in Chapter 6, and a broader formulation was developed for mapping images and video into one dimension and utilizing spectral estimation to detect periodicity. Also, the signals that we choose to analyze are oriented along some angle θ in the spatial frequency domain, which is similar to the Fourier-slice theorem for detection and spatial frequency estimation.

The general framework from Chapter 6 can be developed into numerous applications, and there should be future work to determine the full breadth of inspection technologies that can benefit from this. Future research should also be conducted to determine which parts of an image the HVS will focus on. This will allow us to anticipate inspector inattention, which will assist our algorithm in providing the objective assessment that humans cannot provide.

Overall, the use of signal-processing techniques has produced a robust set of algorithms that should continue to be developed. Such algorithms utilize global information across video frames, spectral estimation for robust periodicity estimation, and a Fourier-slice type of examination in the Fourier domain, which leads to more robust solutions. Future railroad track inspection technology should incorporate automatic detection and segmentation of periodically occurring objects to achieve a more robust system.

REFERENCES

[1] S. Sawadisavi, J. Edwards, E. Resendiz, J. Hart, C. Barkan, and N. Ahuja, “Development of a machine vision system for inspection of railroad track,” in *Proceedings of the American Railway Engineering and Maintenance of Way Association Annual Conference*, 2009.

[2] S. Sawadisavi, J. Edwards, E. Resendiz, J. Hart, C. Barkan, and N. Ahuja, “Machine-vision inspection of railroad track,” in *Proceedings of the Transportation Research Board 88th Annual Meeting*, Washington, DC, 2009.

[3] E. Resendiz, L. Molina, J. Hart, J. Edwards, S. Sawadisavi, N. Ahuja, and C. Barkan, “Development of a machine vision system for inspection of railway track components,” in *Proceedings of the 12th World Conference on Transport Research (WCTR)*, 2010.

[4] J. Hart, E. Resendiz, B. Freid, S. Sawadisavi, C. Barkan, and N. Ahuja, “Machine vision using multi-spectral imaging for undercarriage inspection of railroad equipment,” in *Proceedings of the 8th World Congress on Railway Research (WCRR)*, 2008.

[5] B. Schlake, J. Edwards, J. Hart, C. Barkan, S. Todorovic, and N. Ahuja, “Machine vision condition monitoring of heavy-haul railcar structural underframe components,” in *Proceedings of the International Heavy Haul Conference*, Shanghai, China, 2009.

[6] Y.-C. Lai, C. Barkan, J. Drapa, N. Ahuja, J. Hart, P. Narayanan, C. Jawahar, A. Kumar, L. Milhon, and M. Stehly, “Machine-vision analysis of the energy efficiency of intermodal freight trains,” *Journal of Rail and Rapid Transit*, vol. 221, pp. 353–364, 2007.

[7] J. Edwards, J. Hart, S. Todorovic, C. Barkan, N. Ahuja, Z. Chua, N. Kocher, and J. Zeman, “Development of machine vision technology for railcar safety appliance inspection,” in *Proceedings of the International Heavy Haul Conference, Specialist Technical Session*, 2007, pp. 745–752.

[8] B. Ghanem, E. Resendiz, and N. Ahuja, "Segmentation-based perceptual image quality assessment (SPIQA)," in *International Conference on Image Processing*, 2008, pp. 393–396.

[9] L. Molina, E. Resendiz, J. Edwards, J. Hart, C. Barkan, and N. Ahuja, "Condition monitoring of railway turnouts and other track components using machine vision," in *Proceedings of the Transportation Research Board 90th Annual Meeting*, Washington, DC, 2011.

[10] D. Forsyth and J. Ponce, *Computer Vision: A Modern Approach.* Upper Saddle River, NJ: Prentice Hall, 2002.

[11] B. Manjunathi and W. Ma, "Texture features for browsing and retrieval of image data," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 8, pp. 837–842, 1996.

[12] D. Bailey, "Detecting regular patterns using frequency domain self-filtering," in *IEEE International Conference on Image Processing*, 1997.

[13] G. Bouchard and B. Triggs, "Hierarchical part-based visual object categorization," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2005, pp. 710–715.

[14] R. O. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Transactions on Antennas and Propagation*, vol. 34, no. 3, pp. 276–280, 1986.

[15] S. Marple, *Digital Spectral Analysis.* Englewood Cliffs, NJ: Prentice Hall, 1987.

[16] H. Kim and R. Park, "Extracting spatial arrangement of structural textures using projection information," *Pattern Recognition*, vol. 25, no. 3, pp. 237–245, 1992.

[17] Y. Liu, R. Collins, and Y. Tsin, "A computational model for periodic pattern perception based on frieze and wallpaper groups," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 3, pp. 354–371, 2004.

[18] D. Chetverikov and A. Hanbury, "Finding defects in texture using regularity and local orientation," *Pattern Recognition*, vol. 35, no. 10, pp. 2165–2180, 2002.

[19] F. Liu and R. Picard, "Periodicity, directionality, and randomness: Wold features for image modeling and retrieval," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 7, pp. 722–733, 1996.

[20] J. Orwell, J. Boyce, J. Haddon, and G. Watson, "Detecting periodic structure," in *Proceedings 14th International Conference on Pattern Recognition*, 1998, pp. 714–716.

[21] Y. Keller and Y. Shkolnisky, "A signal processing approach to symmetry detection," *IEEE Transactions on Image Processing*, vol. 15, no. 8, pp. 2198–2207, 2006.

[22] H. Feichtinger and T. Strohmer, *Gabor Analysis and Algorithms.* Boston, MA: Birkhäuser, 1998.

[23] R. Bracewell, "Numerical transforms," *Science*, vol. 248, no. 4956, pp. 697–704, 1990.

[24] E. Resendiz and N. Ahuja, "A unified model for activity recognition in video," in *Proceedings of the 19th International Conference on Pattern Recognition*, 2008, pp. 1–4.

[25] L. Zelnik-Manor and M. Irani, "Event-based analysis of video," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2001, pp. 123–130.

[26] J. Huang, T. Serre, L. Wolf, and T. Poggio, "A biologically inspired system for action recognition," in *IEEE International Conference on Computer Vision*, 2007, pp. 1–8.

[27] P. Dollar, V. Rabaud, G. Cottrell, and S. Belongie, "Behavior recognition via sparse spatio-temporal features," in *IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, 2005, pp. 65–72.

[28] R. T. Collins, R. Gross, and J. Shi, "Silhouette-based human identification from body shape and gait," in *IEEE International Conference on Automatic Face and Gesture Recognition*, 2002, pp. 351–356.

[29] A. Briassouli and N. Ahuja, "Integrated spatial and frequency domain 2D motion segmentation and estimation," in *IEEE International Conference on Computer Vision*, 2005, pp. 244–250.

[30] A. Briassouli and N. Ahuja, "Extraction and analysis of multiple periodic motions in video sequences," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 7, pp. 1244–1261, 2007.

[31] A. V. Oppenheim and R. W. Schafer, *Discrete-Time Signal Processing*, 2nd ed. Upper Saddle River, NJ: Prentice Hall, 1999.

[32] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri, "Action as space-time shapes," in *IEEE International Conference on Computer Vision*, 2005, pp. 1395–1402.

[33] H. R. Sheikh, M. R. Sabir, and A. C. Bovik, “A statistical evaluation of recent full reference image quality assessment algorithms,” *IEEE Transactions on Image Processing*, vol. 15, no. 11, pp. 3440–3451, Nov. 2006.

[34] J. Lubin, *Visual Models for Target Detection and Recognition*, 2nd ed. Singapore: World Scientific, 1995.

[35] Z. Wang, E. P. Simoncelli, and A. C. Bovik, “Multi-scale structural similarity for image quality assessment,” in *Proceedings of the IEEE Asilomar Conference on Signals, Systems, and Computers*, Nov. 2003, pp. 1398–1402.

[36] H. R. Sheikh and A. C. Bovik, “Image information and visual quality,” *IEEE Transactions on Image Processing*, vol. 15, no. 2, pp. 430–444, Feb. 2006.

[37] A. N. Netravali and B. G. Haskell, *Digital Pictures*, 2nd ed. New York, NY: Plenum Press, 1995.

[38] H. Arora and N. Ahuja, “Analysis of ramp discontinuity model for multiscale image segmentation,” in *Proceedings of the International Conference on Pattern Recognition*, vol. 4, Aug. 2006, pp. 99–103.

[39] L. Itti, C. Koch, and E. Neibur, “A model of saliency-based visual attention for rapid scene analysis,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 11, pp. 1254–1259, 1998.