Theory and Applications of
Mathematics & Computer Science

(ISSN 2067-2764, EISSN 2247-6202)
http://www.uav.ro/applications/se/journal/index.php/tamcs

Theory and Applications of Mathematics & Computer Science 4 (2) (2014) 123–139

Fence-like Quasi-periodic Texture Detection in Images

Randima Hettiarachchi[a], James Peters[a,b,*], Neil Bruce[c]

[a]*Department of Electrical & Computer Engineering, University of Manitoba, Winnipeg, MB, R3T 5V6, Canada.*
[b]*Department of Mathematics, Faculty of Arts and Sciences, Adıyaman University, Adıyaman, Turkey.*
[c]*Department of Computer Science, University of Manitoba, Winnipeg, MB, R3T 5V6, Canada.*

Abstract

The focus of this article is on automatic detection of fence or wire mesh (a form of quasi-periodic texture) in images through frequency domain analysis. Textures can be broadly classified in to two general classes: quasi-periodic and random. For example, a fence has a repetitive geometric pattern, which can be classified as a quasi-periodic texture. Quasi-periodic textures can be easily detected in the frequency spectrum of an image as they result in peaks in the frequency spectrum. This article explores a novel way of de-fencing viewed as a quasi-periodic texture segmentation by filtering in frequency domain to segregate the fence from the background. A resulting de-fenced image is followed by support vector machine classification. An interesting application of the proposed approach is the removal of occluding structures such as fence or wire mesh in animal enclosure photography.

Keywords: Frequency spectrum, quasi-periodic texture, texture segmentation
2010 MSC No: Primary 35B36, Secondary 03C13, 03C45, 93C80.

1. Introduction

This article introduces an algorithm to detect automatically fence or wire mesh structures, which typically present in the foreground of the image. A region in an image has a constant texture, provided a set of local statistics or other local properties of the picture function are constant, slowly varying, or approximately periodic (Tuceryan & Jain, 1993). A fence can be classified as a texture in an image. Textures can be broadly classified in to two general classes: *periodic* or more generally *quasi-periodic textures* and *random textures*.

*Corresponding author: 75A Chancellor's Circle, EITC-E2-390, University of Manitoba, WPG, MB R3T 5V6, Canada; e-mail: `james.peters3@ad.umanitoba.ca`, research supported by The Scientific and Technological Research Council of Turkey (TÜBİTAK) Scientific Human Resources Development (BIDEB) under grant no: 2221-1059B211301223 and Natural Sciences & Engineering Research Council of Canada (NSERC) discovery grant 185986.

Email addresses: `hettiarr@cc.umanitoba.ca` (Randima Hettiarachchi), `James.Peters3@umanitoba.ca` (James Peters), `bruce@cs.umanitoba.ca` (Neil Bruce)

According to (Rangayyan, 2004), if there is a repetition of a texture element at almost regular or quasi-periodic intervals, such textures can be classified as quasi-periodic or ordered and the smallest repetitive element is called a texton or a texel. In contrast if no such repetitive element can be identified, those textures can be classified as random.

(Ohm, 2004) classifies textures as *regular* and *irregular* textures. Regular textures refer to textures, which exhibits strong periodic or quasi-periodic behavior. According to (Ohm, 2004), exact periodicity is a very rare case mostly found in synthetic images. The regular structures in natural images are often quasi-periodic, which means that periodic pattern can clearly be recognized, but have slight variations of periods. As it will be shown in section 2, quasi-periodic textures are a generalization of periodic textures.

Based on the above classifications, a fence structure, which has a texture element repeating at quasi-periodic intervals can be categorized as a quasi-periodic texture. Hence, a fence-like texture can be modeled as a quasi-periodic signal, which shows peaks in its power spectrum. It is mentioned in (Chang & Kuo, 1993) that these kinds of quasi-periodic signals possess dominant frequencies located in the middle frequency channels.

The perception of texture has numerous dimensions. Thus, a number of different texture representations were introduced from time to time in order to accommodate a variety of textures. These representations are categorized in (Tuceryan & Jain, 1993) as statistical methods, which involves co-occurrence matrices and autocorrelation features, geometric methods, model based methods and signal processing methods. Signal processing methods are subdivided into spatial domain filtering (Malik & Perona, 1990) and frequency filtering.

Frequency analysis of the textured image is close to human perception of texture as human visual system analyzes the textured image by decomposing the image into its frequency and orientation components (Campbell & Robson, 1968). (Turner, 1986) and (Clark *et al.*, 1987) proposed to use the Gabor filters in texture analysis. The Gabor filter is a frequency and orientation selective filter. Another model, which is widely used for texture analysis is wavelet transform (Chang & Kuo, 1992, 1993; Wilscy & Sasi, 2010).

The focus of this article is on images, which are occluded with fence textures as shown in figure 1. In such cases, it is challenging to segment the fence from the rest of the image, especially when the image background is regular. Simple colour segmentations and edge detection does not work in this case.

The traditional frequency filters used for texture analysis, Gabor and Wavelet cannot be directly applied to extract fence texture in our scenario as the frequencies correspond to both fence and the background are present in the spectrum. Thus, we first perform frequency domain processing to isolate fence texture from the background and subsequently apply Wavelet transform.

An interesting application of the proposed algorithm can be detection and removal of fence-like textures obstructing the images in zoo photography. According to many web articles on photography (Stalking, 2010; Masoner, 2013), wire mesh and fences are a major challenge in zoo photography. The algorithm proposed in this article was tested for fences with different shapes, sizes, colours and orientations.

The rest of the article is organized as follows. Section 2 introduces quasi-periodic signals and provides the mathematical background to analyze quasi-periodic signals in images. Section 3 discusses the implementation of the quasi-periodic texture detection algorithm in three steps: (1)

frequency domain filtering for quasi-periodic texture detection, (2) multiresolution processing for fence mask formation and (3) fence segmentation through SVM classification. The experimental results of the proposed algorithm are given in Section 4 for some zoo images as well as for some challenging images from PSU NRT Database (Liu, 2007). A comparison of the proposed method with existing fence detection techniques is given in section 5 followed by future work and conclusion in sections 6 and 7 respectively.

Figure 1. Images Occluded with Fence Textures.

2. Quasi-periodic Signals

Before going into details of quasi-periodic texture detection in images, understanding the mathematical background of quasi-periodic signals is important.

Definition 2.1. Continuous-time Periodic Signal ((Proakis & Manolakis, 2006, §1, p. 13))
By definition, A continuous signal f(t) locally defined on the set $L^2(\mathfrak{R})$ of finite energy signals is fully periodic with period T, when the signal exactly satisfies

$$f(t) = f(t + T).$$

Definition 2.2. Continuous-time Quasi-periodic Signal ((Martin *et al.*, 2010))
A signal $f_{qp}(t)$ is quasi-periodic with k periods $T_1, ..., T_k$ when

$$f_{qp}(t) = g\{f_1(t), f_2(t), ..., f_k(t)\},$$

where the k signals $f_i(t)$ are continuous periodic signals with respect to each period T_i.

In the case of continuous functions locally defined on the set $L^2(\mathfrak{R})$ of finite energy signals, quasi-periodic signals are a generalization of periodic signals. All the periods are required to be strictly positive and to be rationally linearly independent (Martin *et al.*, 2010).

Definition 2.3. Discrete-time Periodic Signal((Proakis & Manolakis, 2006, §1, p. 15))
A discrete-time signal f(n) is periodic with period N, if and only if,

$$f(n) = f(n + N) \text{ for all n.}$$

Based on the definition of continuous-time quasi-periodic signals, the definition for discrete-time quasi-periodic signals can be derived.

Definition 2.4. Discrete-time Quasi-periodic Signal
A discrete-time signal $f_{qp}(n)$ is quasi-periodic with k periods $N_1, ...N_k$ when

$$f_{qp}(n) = g\{f_1(n), f_2(n), ..., f_k(n)\},$$

where $g : \mathbb{Z}^k \to \mathbb{Z}$ and the k signals $f_i(n)$ are discrete-time periodic signals with respect to each period N_i.

In the context of this paper, an image is considered as a 2D discrete-time signal. If we extend the definition of 1D quasi-periodic signal to 2D quasi-periodic signal;

Definition 2.5. 2D Discrete-time Periodic Signal((Woods, 2006, §1, p. 7))
A 2D discrete-time signal f(x,y) is periodic with period (M,N), if and only if,

$$f(x, y) = f(x + M, y) = f(x, y + N), \forall n, m \in \mathbb{Z}.$$

Definition 2.6. 2D Discrete-time Quasi-periodic Signal
A 2D discrete-time signal $f_{qp}(x, y)$ is quasi-periodic with k periods $(M_1, ...M_k, N_1, ...N_k)$ when

$$f_{qp}(x, y) = g\{f_1(x, y), f_2(x, y), ..., f_k(x, y)\},$$

where the k signals $f_i(x, y)$ are discrete-time periodic signals with respect to periods (M_i, N_i). Hence, a quasi-periodic signal can be defined as a combination of periodic signals with incommensurate (not rationally related) frequencies (Battersby & Porta, 1996). If the frequencies are commensurate, then f_{qp} becomes a periodic signal (Regev, 2006).

A discrete-time quasi-periodic signal can be expressed with a Fourier series as given in definition 2.8 as a generalization of definition 2.7. 1D case will be considered for simplicity and it can be extended to 2D.

Definition 2.7. Fourier Series of a Discrete-time Periodic Signal ((Proakis & Manolakis, 2006, §4, p. 242))

$$f(n) = \sum_{k=0}^{N-1} c_k exp\left(\frac{j2\pi kn}{N}\right).$$

Definition 2.8. Fourier Series of a Discrete-time Quasi-periodic Signal ((Regev, 2006, p. 156))
The Fourier series of a r-quasi-periodic signal is given by (Regev, 2006):

$$f_{qp}(n) = \sum_{k_1}\sum_{k_2} ... \sum_{k_r} c_{k_1 k_2 ... k_r} exp\left[j\left(\frac{2\pi k_1 n}{N_1} + \frac{2\pi k_2 n}{N_2} + ... + \frac{2\pi k_r n}{N_r}\right)\right],$$

where k=1,2,...,r and the frequencies $\omega_k = 2\pi/N_k$ are incommensurate.

Theorem 2.1. *Let $f_{qp}(n)$ be a discrete-time quasi-periodic signal. Then the frequency spectrum of $f_{qp}(n)$ consists of a set of peaks determined by the fundamental frequencies of each discrete periodic signal component in the signal.*

Proof. With $\omega_i = 2\pi/N_i$, $f_{qp}(n)$ in definition 2.8 can be re-written as

$$f_{qp}(n) = \sum_K c_K exp\left[jK\Omega n\right],$$

where $K = (k_1, k_2, ..., k_r)$ and $\Omega = (\omega_i, \omega_2, ..., \omega_r)$. Thus, the frequency spectrum contains numerous peaks at all frequencies ν, satisfying

$$2\pi\nu = |K \cdot \Omega| = |k_1\omega_1 + k_2\omega_2 + ... + k_r\omega_r|,$$

for any combination of integers $k_1, k_2, ..., k_r$. □

3. Quasi-periodic Texture Detection in Frequency Domain

3.1. Frequency Domain Filtering for Quasi-periodic Texture Detection

As proven by theorem 2.1, the Fourier spectrum of a quasi-periodic signal consists of a discrete set of spikes or peaks at a number of frequencies depending on the number of periodic signals it is comprised of. Hence, based on theorem 2.1, the fence-like quasi-periodic structure should result in peaks in the frequency spectrum of the image. The objective of this section is to filter those spikes in the frequency spectra relevant to the quasi-periodic signal in order to extract the fence texture corresponding to the quasi-periodic signal from the rest of the image.

To achieve this, first start with the frequency domain representation of the 2D image. We will be considering the DFT of an image.

$$F(u,v) = \sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)exp\left[-j2\pi\left(\frac{ux}{M} + \frac{vy}{N}\right)\right] \text{u=0,1,...M-1, v=0,1,...N-1.} \tag{3.1}$$

To filter the frequencies showing spikes in the frequency spectra, it is necessary to perform thresholding based on the magnitude of each frequency component. A filter function $H_1(u,v)$ in frequency domain can be defined for this purpose as given below.

$$H_1(u,v) = \begin{cases} 1 & \text{if } |F(u,v)| > T, \\ 0 & \text{otherwise,} \end{cases} \tag{3.2}$$

where T is a threshold to filter spikes in frequency.

Once the thresholding is applied to the frequency components:

$$F'(u,v) = H_1(u,v)F(u,v)$$

Although, we filtered the frequency components corresponding to peaks in the frequency spectra, it is necessary to filter peaks in frequencies resulted by other details in the image. For an example, the DC component F(0,0), which can be derived by substituting u=0 and v=0 in equation 3.1. $|F(0,0)|$ typically is the largest component of the spectrum.

$$F(0,0) = MN\frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y) = MN\overline{f}(x,y).$$

The quasi-periodic signal in our case is the fence. Fence-like textures typically result in qausi-periodic signals whose dominant frequencies are located in the middle frequency channels (Chang & Kuo, 1993). Therefore, by using a bandpass filter in frequency domain, the frequencies corresponding to the fence can be filtered.

$$H_2(u,v) = \begin{cases} 1 & \text{if } D_1 \leq D(u,v) \leq D_2, \\ 0 & \text{otherwise.} \end{cases} \tag{3.3}$$

where D_1 and D_2 are constants and D(u,v) is the distance between a point (u,v) in the frequency domain and the center of the frequency spectrum.

Thus, the final result in frequency domain after applying the second filter would be:

$$\begin{aligned} F''(u,v) &= H_2(u,v)F'(u,v), \\ &= H_2(u,v)H_1(u,v)F(u,v), \\ &= H(u,v)F(u,v), \end{aligned}$$

where $H = H_2H_1$, since the application of H_1 and H_2 can be considered as a cascade system. When $F''(u,v)$ is transferred back into spatial domain, the resulting image is given by:

$$g(x,y) = \frac{1}{MN}\sum_{u=0}^{M-1}\sum_{v=0}^{N-1} F''(u,v)exp\left[j2\pi\left(\frac{ux}{M}+\frac{vy}{N}\right)\right] \text{x=0,1,...M-1, y=0,1,...N-1.}$$

It is important to note that H_1 and H_2 are zero phase shift filters, which affect the magnitude of the frequency spectra, but do not alter the phase angle. These filters affect the real (Re(u,v)) and imaginary (Im(u,v)) parts equally, thus cancels out when calculating phase angle $\phi(u,v) = arctan[Im(u,v)/Re(u,v)]$.

Figure 2(d) illustrates the final result of frequency domain filtering explained above. It can be clearly seen that the fence texture is emphasized and other image details have been suppressed.

3.2. *Multiresolution Processing for Fence Mask Formation*

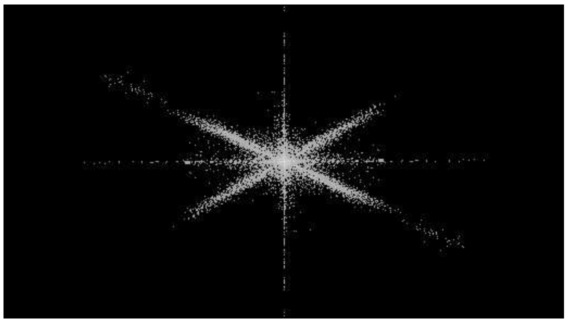
The human visual system analyzes the textured images by decomposing the image into its frequency and orientation components (Campbell & Robson, 1968). Wavelet transformation provides the ability to analyze images through multiresolution processing.

Wavelet transform in two dimension provides the two dimensional scaling function $\phi(x,y)$ and three two dimensional directionally sensitive wavelets $\psi^H(x,y)$, $\psi^V(x,y)$,$\psi^D(x,y)$ as given in (Gonzalez & Richard, 2002).

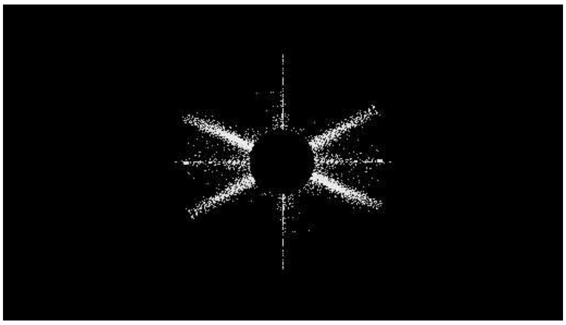
$$\phi_{j,m,n}(x,y) = 2^{\frac{j}{2}}\phi(2^jx - m, 2^jy - n).$$

$$\psi^i_{j,m,n}(x,y) = 2^{\frac{j}{2}}\psi^i(2^jx - m, 2^jy - n), i = \{H,V,D\}.$$

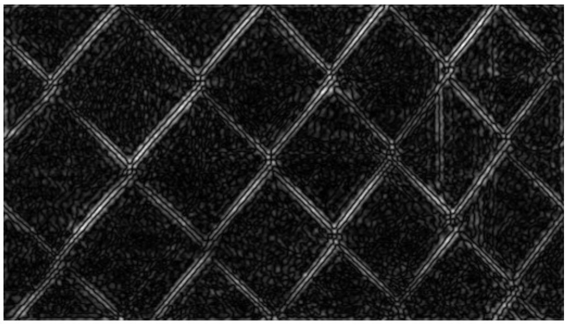
(a) Original Image (f(x,y)).



(b) Filtered Peak Frequencies (F'(u,v)).



(c) Bandpassed Spectrum (F"(u,v)).



(d) Spatial Domain (g(x,y)).

Figure 2. Frequency Domain Filtering for Fence Texture Segregation from Image Background.

These wavelets measure intensity variations for images along different directions: ψ^H measures variations along horizontal direction (along columns), ψ^V measures variations along vertical direction (along rows) and ψ^D corresponds to variations along diagonals.

The discrete transform of image f(x,y) is:

$$W_\phi(j_0, m, n) = \frac{1}{\sqrt{MN}} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y)\phi_{j_0,m,n}(x, y).$$

$$W_\psi^i(j, m, n) = \frac{1}{\sqrt{MN}} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y)\psi^i_{j,m,n}(x, y), i = \{H, V, D\},$$

where j_0 is an arbitrary starting scale and the $W_\phi(J_0, m, n)$ coefficients define an approximation of f(x,y) at scale j_0. The $W_\psi^i(j, m, n)$ coefficients add horizontal, vertical and diagonal details for scales $j \geq j_0$. $W_\psi^i(j_0, m, n)$ coefficients are called detail coefficients. Usually j_0 is set to zero.

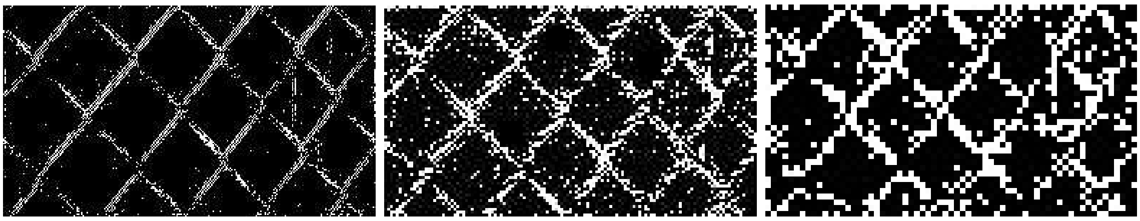
For each level j, thresholding is performed on the details coefficients $W_\psi^i(j, m, n)$ to extract the fence masks $M^i(j, m, n)$ at each level j.

$$M^i(j, m, n) = \begin{cases} 1 & \text{if } W_\psi^i(j, m, n) > T_j, \text{ where } T_j \text{ is the threshold for level } j, \\ 0 & \text{otherwise.} \end{cases}$$

The final fence mask at level j is obtained by performing *OR* operation of the vertical, horizontal and diagonal fence masks at level j.

$$M(j, m, n) = M^V(j, m, n) \oplus M^H(j, m, n) \oplus M^D(j, m, n).$$

The detected fence masks at 3 consecutive levels are shown in figure 3.



(a) M(1,m,n). (b) M(2,m,n). (c) M(3,m,n).

Figure 3. Detected Fence Masks at Three Different Levels.

Next, the fence masks at different levels of wavelet pyramid were combined by using a coarser to finer strategy. The objective is to reduce noise and extract pixels, which fall exactly on the fence. In order to make the resultant mask in the same size as the original image, a mask was created at the zero level by just thresholding the spatial domain result of frequency filtering (g(x,y)). Hence, altogether we have fence masks at 4 different levels in the pyramid.

First, the highest level fence mask (level 3) was considered and if a pixel belongs to the mask then we move to the next lower level (level 2) and check for the neighbouring children of the original pixel. If any of the neighbouring children are mask pixels, then recursively go and check for their neighbouring children in the subsequent lower levels. Finally, when the algorithms reaches the bottom most level (zero level), it marks the mask pixels as 1, given that the neighbouring children in the lowest level are mask pixels as well. The resultant fence mask is shown in Figure 4.

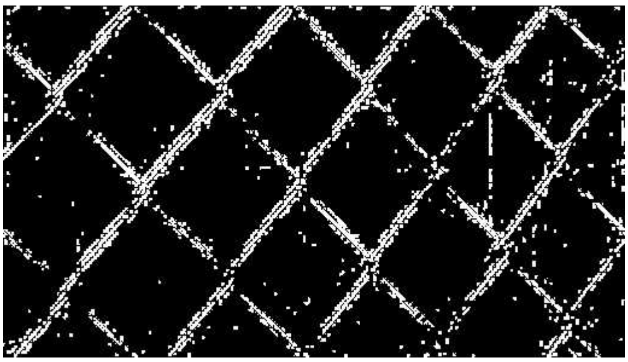


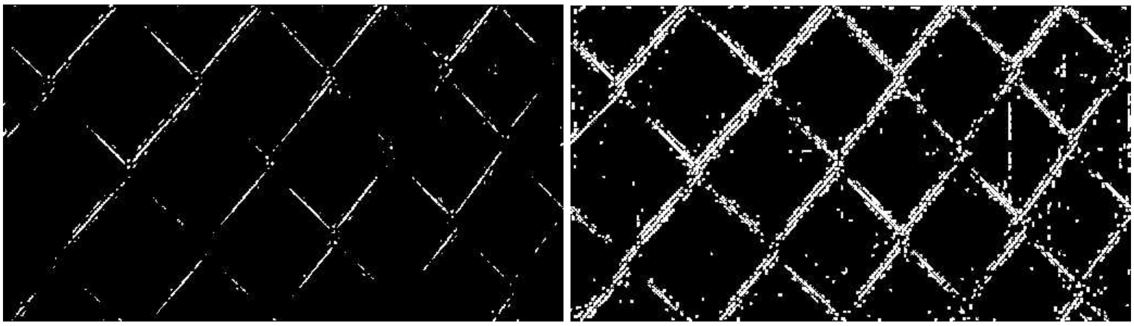
Figure 4. Fence Mask Formed by Combining Wavelet Decomposition Levels.

3.3. Fence Segmentation through SVM Classification

Although the noise is minimized and the fence is emphasized in the detected fence mask, it is not perfectly detected yet. However, the detected fence mask classifies a good number of pixels, which exactly falls on the fence in the image. This knowledge on fence pixels can be used to segment the fence. Hence, it was decided to pick some samples from the fence mask and use the features of those sample pixels to train a *Support Vector Machine (SVM) classifier* in order to segment the fence texture. A SVM classifier with a linear kernel is used in this case.

In addition to the samples from fence, it is necessary to pick samples from background to train the SVM classifier. For this purpose two root level fence masks were generated. One root level mask was generated by selecting a very high threshold and the other one is generated by using a very low threshold. These masks were used as the root level mask in the process of combining wavelet decomposition levels as explained in section 3.2 separately in order to generate two different final fence masks as shown in Figure 5.

As it can be clearly seen, the root level mask with high threshold generates a very thin final mask, resulting points, which exactly lie on the fence. On the other hand the root level mask with low threshold generates a thick fence mask, which has some points fall on the background as well.



(a) Thin Mask with High Threshold. (b) Thick Mask with Low Threshold.

Figure 5. Two Fence Masks used for SVM Classification.

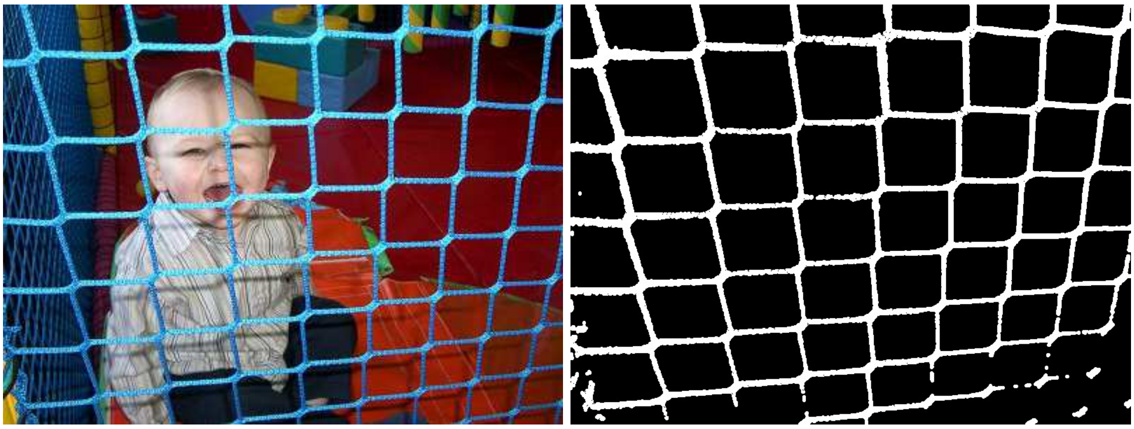
The thin mask was used to pick random samples, which represent fence class and the negation of the thick mask (1-thick mask) is used to pick random samples, which represent the background class. The use of negation of thick mask for background sample selection reduces the chance of picking fence pixels as background pixels and hence improves the accuracy of classification.

The feature vector selected for classification plays a very important role in this case as it affects the overall performance of the classification. The RGB colour channels and the gradient direction of the samples were used as the feature set for classification. The resultant fence mask can be further improved with the help of morphological operations.

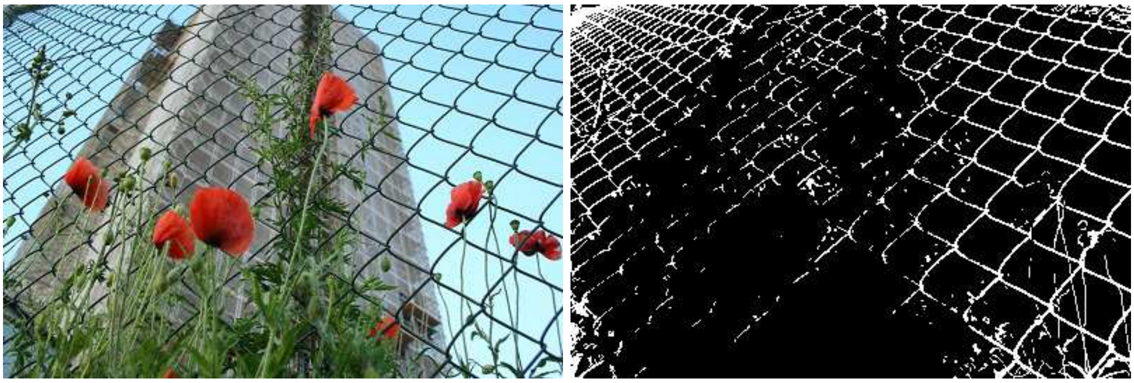
The algorithm to achieve fence-like quasi-periodic texture detection in digital images is given in Algorithm 1.

Algorithm 1 Algorithm for fence-like quasi-periodic texture detection in images

```
1: Read the fenced image I
2: Convert I into frequency domain using Discrete Fourier Transform (let the output be F)
3: Filter F using the peak frequency filter H_1 defined in equation 3.2 (let the output be F1)
4: Filter F1 using the band pass filter H_2 defined in equation 3.3 (let the output be F2)
5: Convert F2 back into spatial domain (let the output be filtI)
6: Perform Wavelet decomposition on filtI with three decomposition levels
7: for each Wavelet decomposition level do
8:     Find vertical (V), horizontal (H) and Diagonal (D) components
9:     Threshold V, H and D with the same threshold
10:    Combine thresholded V, H and D components using logical OR operation
11: end for
       ▹ %comment: Obtain fence mask by combining all three levels of the wavelet pyramid (let
    the output be fenceMask)%
12: Start from the highest Wavelet decomposition level (level 3)
13: for each pixel in level 3 do
14:     if a pixel belongs to the mask then
15:         Move to next lower level
16:         if current level == lowest level then
17:             Mark the pixel as mask pixels
18:             Mark the neighbouring children as mask pixels
19:         else
20:             Check neighbouring children
21:             if neighbouring children are mask pixels then
22:                 Go back to step 14
23:             end if
24:         end if
25:     end if
26: end for
27: Prepare the training data matrix using feature vectors of sample pixels fall on fence (fence-
    Mask==1) and background (fenceMask==0).
28: Train the SVM classifier by using training data matrix of step 25.
29: Perform SVM classification by using the trained classifier in step 26 by giving original image
    as the input to obtain final fence mask.
```



(a) Child Image. (b) Fence Mask for Child Image.



(c) Building Image. (d) Fence Mask for Building Image.

(e) Flower Image. (f) Fence Mask for Flower Image.

Figure 6. Results of Fence-like Texture Detection in Images from PSU NRT Database (Liu, 2007).

4. Experimental Results

The frequency domain-based fence-like quasi-periodic texture detection algorithm proposed in this article was implemented in Matlab R2013a and it was tested with a number of images with fence-like texture. Some test images were obtained from PSU Near-regular Texture database (Liu, 2007). Images with fences of different shapes (square and diagonal), sizes, colours and orientations were used for this experiment. Figure 6 illustrates results of some of the challenging cases encountered during experiments.

For the completion of the sample application chosen in this paper, once the fence texture was successfully detected and removed, the region, which belonged to the fence, should be filled with relevant information in order to obtain the final image. One of the techniques, which can serve this purpose is *inpainting*. According to (Bertalmio *et al.*, 2000), inpainting is the *modification of images in a way that is non-detectable for an observer who does not know the original image*. There are numerous inpainting techniques introduced in past literature.

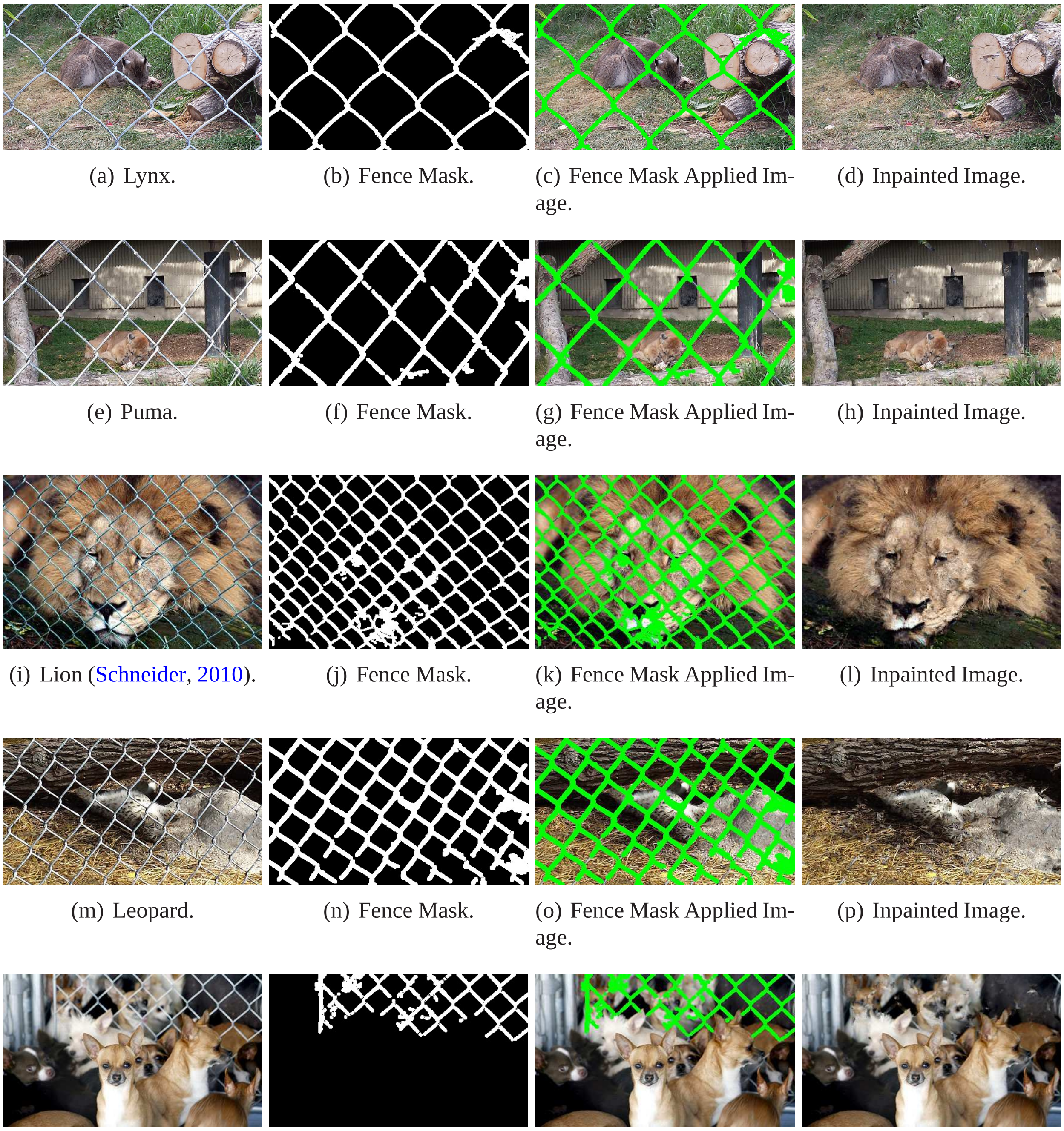
For examples region filling and object removal by exemplar-based image inpainting by Criminisi et al. (Criminisi *et al.*, 2004), Fields of experts by Roth et al. (Roth & Black, 2009) and Image completion with structure propagation by Sun et al. (Sun *et al.*, 2005). Among these techniques, the exemplar based image inpainting technique (Criminisi *et al.*, 2004) was used to fill the fence region in this approach. The results are given in figure 7.

Interestingly, some image distortions can be observed after performing inpainting for some images. The region belonged to the fence texture is much more difficult to texture fill than large, circular regions of similar area. The fence texture in this case is usually wide spread in the whole image. Thus, it requires the inpainting algorithm to correctly propagate and join different types of structures in order to fill this wide spread fence region. Hence, mistakes in structure propagation can be quiet frequent in this case. The high ratio of foreground area to background area and the fragmented background source textures may become challenging for the inpainting technique.

5. Comparison with Existing Fence Detection Techniques

Most of the articles, which investigated the image de-fencing problem, have used a texture based approach to detect the fence, based on the assumption that a fence is a near regular structure. (Liuy *et al.*, 2008) introduced an image de-fencing technique based on lattice structure of regular textures in their article. The de-fencing algorithm proposed in (Liuy *et al.*, 2008) consists of three steps. *(1) automatically finding the skeleton structure of a potential frontal layer in the form of a deformed lattice; (2) classifying pixels as foreground or background using appearance regularity as the dominant cue, and (3) inpainting the foreground regions using the background texture which is typically composed of fragmented source regions to reveal a complete, non-occluded image* (Liuy *et al.*, 2008).

In the first step, to automatically detect the lattice of the fence, (Liuy *et al.*, 2008) uses the iterative algorithm explained in (Hays *et al.*, 2006), which tries to find the most regular lattice for a given image by assigning the neighbour relationships such that neighbors have maximum visual similarity. Step one results in a mesh of quadratiles, which contains repeated elements or texels. In the second step standard deviation of each colour channel and the color features are used for k-means clustering for background foreground separation. In order to obtain the standard deviation,



(a) Lynx. (b) Fence Mask. (c) Fence Mask Applied Image. (d) Inpainted Image.

(e) Puma. (f) Fence Mask. (g) Fence Mask Applied Image. (h) Inpainted Image.

(i) Lion (Schneider, 2010). (j) Fence Mask. (k) Fence Mask Applied Image. (l) Inpainted Image.

(m) Leopard. (n) Fence Mask. (o) Fence Mask Applied Image. (p) Inpainted Image.

(q) Puppies (Liu, 2007). (r) Fence Mask. (s) Fence Mask Applied Image. (t) Inpainted Image.

Figure 7. Results of Fence Removal from Zoo Images.

the texels were aligned and arranged in a stack and standard deviation is calculated along each vertical column of pixels. Finally, texture based inpainting technique introduced by Criminisi et al. (Criminisi *et al.*, 2004, 2003) is used to obtain the final de-fenced image.

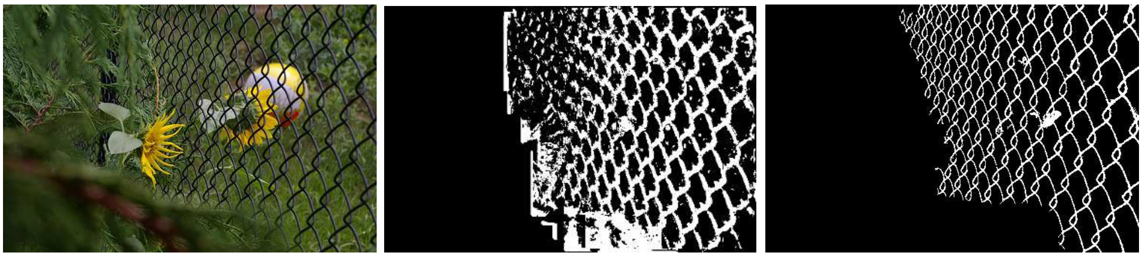
Park et al. revisits the image de-fencing problem in their paper (Park *et al.*, 2011). They no longer uses the lattice detection algorithm introduced in (Hays *et al.*, 2006), as they states its performance is far from practical due to inaccuracy and slowness. Rather the implementation of lattice detection algorithm in (Park *et al.*, 2011) is similar to (Park *et al.*, 2009). In their method, once the type of the repeating pattern is learnt, the irregularities are removed and the learned regularity is used in evaluating the foreground appearance likelihood during the lattice growth. They have improved the lattice detection algorithm by introducing an online learning and classification.

In essence, the de-fencing algorithms introduced in both of these articles uses a lattice detection algorithm in order to find the fence mask. Thus, the success of both algorithms depends on finding the repeated element or texel in the fence structure. The lattice detection algorithm used by (Liuy *et al.*, 2008) has no measures against irregularities in the lattice while the lattice detection algorithm used by (Park *et al.*, 2011) takes some measures to remove irregularities during lattice growth. However, both these approaches depend on the regularity of the fence as well as the irregularity of the background of the image. Although (Park *et al.*, 2011) takes measures against irregularities in the fence, it does not take in to account the possibility of regularities in the background. Furthermore, the lattice detection process itself is very complex and time consuming.

In contrast to the two methods discussed above, the method explained in this article uses a frequency domain approach to address the fence detection problem. Due to the uncertainty principle, the global wide spread fence texture in spatial domain becomes local to a set of frequencies in the frequency domain. So the processing required to extract the fence texture in frequency domain is simpler and faster compared to spatial domain processing. This becomes advantageous in the proposed method compared to the existing techniques. Moreover, the band pass filtering in frequency domain used in the proposed method helps to avoid other periodic structures (regularities) in the background, which is not possible in existing techniques. The proposed method is robust against deformations and irregularities in the fence texture due to SVM classification used in fence segmentation phase.

The existing near regular lattice detection approaches work well for some images and on the other hand fail for some cases. They have observed that the failure cases are often accompanied by sudden changes of colors in the background and obscuring objects in front of the fence. For examples in (Liuy *et al.*, 2008) method, the lattice detection fails for images (a) and (c) in Figure 6 and for image (q) in Figure 7. The proposed method is successful in detecting fence texture in all those images. A comparison of fence mask detected in Flower image by (Liuy *et al.*, 2008) method and proposed method is given in Figure 8.

However, the proposed method fails to provide satisfactory results for blurred images, especially when the fence is very much blurred. In such cases preprocessing to sharpen the fence may give better results. Furthermore, fence segmentation becomes challenging when the visual similarity between fence pixels and background pixels becomes high. Feature set used for segmentation has to be tuned to overcome such problems. Determining the correct feature set is challenging in such scenarios.



(a) Flower Image. (b) (Liuy *et al.*, 2008). (c) Proposed Method.

Figure 8. Comparison of Fence Mask Detected for Flower Image.

6. Future Work

Fence texture segmentation becomes challenging, when there are pixels with features similar to fence pixels in the background. SVM classification used for final segmentation of the fence texture in this article can be replaced with descriptive motif pattern generation described in (Peters & Hettiarachichi, 2013). The accuracy of this phase can be further improved with help of near set theory (Peters, 2013; Peters & Naimpally, 2012; Peters, 2014; Peters *et al.*, 2014).

7. Conclusion

Fence-like texture present in the foreground of the image occludes the points of interest in an image and is difficult to segment by directly applying conventional frequency filters used for texture analysis. The proposed approach in this article segregates each fence texture by frequency domain processing prior to wavelet transformation and the segmentation is achieved through support vector machine classification.

The proposed method works well for fence texture with different shapes, sizes, colours and orientations. Fence texture detection was successful not only for images having fence in the foreground but also for images having fence in the background.

As a sample application of the proposed approach, removal of fences from zoo animal enclosure images is presented. In addition to this, the proposed approach to de-fencing can be used for any application, where the images are occluded with fence-like texture.

References

Battersby, Nicholas C and Sonia Porta (1996). *Circuits and systems tutorials*. Wiley. com.

Bertalmio, Marcelo, Guillermo Sapiro, Vincent Caselles and Coloma Ballester (2000). Image inpainting. In: *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*. ACM Press/Addison-Wesley Publishing Co.. pp. 417–424.

Campbell, Fergus W and JG Robson (1968). Application of fourier analysis to the visibility of gratings. *The Journal of Physiology* **197**(3), 551.

Chang, Tianhorng and C-CJ Kuo (1992). A wavelet transform approach to texture analysis. In: *Acoustics, Speech, and Signal Processing, 1992. ICASSP-92., 1992 IEEE International Conference on*. Vol. 4. IEEE. pp. 661–664.

Chang, Tianhorng and C-CJ Kuo (1993). Texture analysis and classification with tree-structured wavelet transform. *Image Processing, IEEE Transactions on* **2**(4), 429–441.

Clark, Marianna, Alan C Bovik and Wilson S Geisler (1987). Texture segmentation using gabor modulation/demodulation. *Pattern Recognition Letters* **6**(4), 261–267.

Criminisi, Antonio, Patrick Perez and Kentaro Toyama (2003). Object removal by exemplar-based inpainting. In: *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*. Vol. 2. IEEE. pp. II–721.

Criminisi, Antonio, Patrick Pérez and Kentaro Toyama (2004). Region filling and object removal by exemplar-based image inpainting. *Image Processing, IEEE Transactions on* **13**(9), 1200–1212.

Gonzalez, Rafael C and E Richard (2002). Digital Image Processing. *ed: Prentice Hall Press, ISBN 0-201-18075-8*.

Hays, James, Marius Leordeanu, Alexei A Efros and Yanxi Liu (2006). Discovering texture regularity as a higher-order correspondence problem. In: *Computer Vision–ECCV 2006*. pp. 522–535. Springer.

Liu, Yanxi (2007). PSU near-regular texture database. http://vivid.cse.psu.edu/texturedb/gallery/.

Liuy, Yanxi, Tamara Belkina, James H Hays and Roberto Lublinerman (2008). Image de-fencing. In: *Proc. IEEE Conf. Computer Vision and Pattern Recognition*. pp. 1–8.

Malik, Jitendra and Pietro Perona (1990). Preattentive texture discrimination with early vision mechanisms. *JOSA A* **7**(5), 923–932.

Martin, Nadine, Corinne Mailhes et al. (2010). About periodicity and signal to noise ratio-the strength of the autocorrelation function.. In: *Seventh International Conference on Condition Monitoring and Machinery Failure Prevention Technologies. CM 2010 and MFPT 2010, Stratford-upon-Avon, UK, 22-24 June 2010.*

Masoner, Liz (2013). How to take great zoo photos http://photography.about.com/od/animalphotography/a/zoophotos.htm.

Ohm, Jens R (2004). *Multimedia communication technology: Representation, transmission and identification of multimedia signals*. Springer.

Park, Minwoo, Kyle Brocklehurst, Robert T Collins and Yanxi Liu (2009). Deformed lattice detection in real-world images using mean-shift belief propagation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* **31**(10), 1804–1816.

Park, Minwoo, Kyle Brocklehurst, Robert T Collins and Yanxi Liu (2011). Image de-fencing revisited. In: *Computer Vision–ACCV 2010*. pp. 422–434. Springer.

Peters, J.F. (2013). Local near sets: Pattern discovery in proximity spaces. *Math. in Comp. Sci.* **7**(1), 87–106. doi: 10.1007/s11786-013-0143-z.

Peters, J.F. (2014). *Topology of Digital Images. Visual Pattern Discovery in Proximity Spaces*. Vol. 63 of *Intelligent Systems Reference Library*. Springer. ISBN 978-3-642-53844-5, pp. 1-342.

Peters, J.F. and R. Hettiarachichi (2013). Visual motif patterns in separation spaces. *Theory and Applications of Mathematics & Computer Science* **3**(2), 36–58.

Peters, J.F. and S.A. Naimpally (2012). Applications of near sets. *Notices of the Amer. Math. Soc.* **59**(4), 536–542. DOI: http://dx.doi.org/10.1090/noti817.

Peters, J.F., E. İnan and M.A. Öztürk (2014). Spatial and descriptive isometries in proximity spaces. *General Mathematics Notes* **21**(2), 1–10.

Proakis, John and Dimitris Manolakis (2006). *Digital Signal Processing: Principles, Algorithms and Applications*. Prentice Hall.

Rangayyan, Rangaraj M (2004). *Biomedical image analysis*. CRC press.

Regev, Oded (2006). *Chaos and complexity in astrophysics*. Cambridge University Press.

Roth, Stefan and Michael J Black (2009). Fields of experts. *International Journal of Computer Vision* **82**(2), 205–229.

Schneider, Mara Kay (2010). African adventures at the zion wildlife gardens. http://maerchens-adventures.blogspot.ca/2010/08/african-adventures-at-zion-wildlife.html.

Stalking, Light (2010). The three main challenges of zoo photography (and how to overcome them). http://www.lightstalking.com/zoo-photography-challenges.

Sun, Jian, Lu Yuan, Jiaya Jia and Heung-Yeung Shum (2005). Image completion with structure propagation. *ACM Transactions on Graphics (ToG)* **24**(3), 861–868.

Tuceryan, Mihran and Anil K Jain (1993). Texture analysis. *Handbook of pattern recognition and computer vision*.

Turner, Mark R (1986). Texture discrimination by gabor functions. *Biological Cybernetics* **55**(2-3), 71–82.

Wilscy, M and Remya K Sasi (2010). Wavelet based texture segmentation. In: *Computational Intelligence and Computing Research (ICCIC), 2010 IEEE International Conference on*. IEEE. pp. 1–4.

Woods, John W (2006). *Multidimensional signal, image, and video processing and coding*. Academic press.