A Blind Blur Detection Method for Electro-optic (EO) Images

Gregory Bower[1]

[1]QorTek, Inc, Williamsport, PA, 17701, USA
gbower@qortek.com

ABSTRACT

Blurring in Electro-Optic (EO) images is a significant issue that can arise due to the payload and platform operations. It would be advantageous for unmanned platforms to determine if significant blurring is present within captured images before the images are observed and the collection sequence has ended. In this way, the degradation can be identified and remedied in operation in real-time. In this paper, we demonstrate that a statistical algorithm called Symbolic Analysis (SA) is suitable for detecting blurring in the output images of EO systems. The SA algorithm adapted from previous work is described and demonstrated on an example image with artificial Gaussian-based blurring induced.

1. INTRODUCTION

The first thoughts that come to mind when one talks about diagnostics tends to either be in the medical mindset or perhaps the current state of an operating machine or system. Diagnostics means the “practice or technique of diagnosing” which can be applied to many systems in many situations. Systems that generally wear out can be diagnosed and tracked though time in order to achieve a prognostic measure of sorts in order to determine when maintenance or repair to a system is deemed necessary. A diagnostic measure does not necessarily represent a wear out mechanism in the system but instead a measure of when the system is not operating as intended an example of which is when a system is improperly setup.

In this paper, we apply a diagnostic algorithm to an electro-optic (EO) system. EO systems are typically employed in surveillance or reconnaissance platforms and are used to keep track of scenes or targets. In general, an EO system does not typically wear out as would mechanical or other electrical systems. Indeed, it is the case that these EO systems tend to employ a DSLR type of camera which after a certain amount of shutter actuations, the shutter itself can fail. This type of failure does not typically induce a signature within an image; instead it is simply tracked by total number of exposures taken by the camera. Once this limit is reached, the camera is replaced or shutter is replaced.

All is not lost in the application of diagnostic algorithms to EO systems. Indeed, the quality of the photo can be used as the diagnosis target. A critical feature of photos that must be maintained in high resolution EO images is sharpness meaning blurring within the image must be minimized. An image with minimal blurring provides the highest possible feature resolution. Burring within an image cannot be determine until the image is processed either onboard or by downloading the image from the platform. Otherwise, the degraded images are not identified until after all the images have been downloaded from the DSLR or another camera payload. Image blurring can be a result from either platform motion blur and/or by focusing blur. Focusing blur is due to the camera having an improper focal setting and motion blur is due to the camera movement.

In this regard, in order to produce the best possible image, it would be ideal to identify blurring in images during image capture so that actions can be taken in order to improve the quality of the captured images. In this scenario, an algorithm or process is needed that can quickly and efficiently process an image and identify if blurring is present in the image. The issue is determining what constitutes blurring in an image taken when the scenery is unknown from an image processing standpoint. This is the so-called blind identification (Abed-Meraim, 1997). Blind identification is obtaining relevant information from a system utilizing only its output data.

Blind blur detection has been a recent research topic and is gaining interest as more payloads and systems become autonomous. A general overview of blur detection algorithms can be found in (Koik, 2013). The approaches covered in this paper include edge detection algorithms, Bayes approaches, and wavelet methods (Tong, 2004). Wavelet methods allow for separation of ‘image frequency’ so that detailed analysis can be carried out on the image. Once decomposed, the data can be processed to determine if certain high frequency edges are contained in the image. Image blurring is likely if these edges are missing. In general, the idea is to form some measure on the amount of high (or low) frequency content within a digital image.

Our approach utilizes statistics in order to determine the blurring contained in the image. A similar approach to blind blur detection is given in (Tiwari, 2014). In (Tiwari, 2014), the authors propose using statistics of image phase congruency and the magnitude information of the image in order to detect blurring. The phase congruency information is extracted from Fourier analysis of the image and this is coupled with the image's magnitude information.

The objective is to be able to quantify some extent of blurring in an image irrespective of the scenery in which the EO system operates. This is challenging as a uniform scene such as a field or airport can produce drastically different measures then those obtained from scenes containing trees and foliage. Our approach is to model these differences statistically and use the statistical information to infer the amount of blurring in an image. The algorithm implemented was called the Symbolic Analysis (SA) algorithm (Bower, 2015).

The SA algorithm was investigated to determine if it is an appropriate and feasible approach to detecting blurring in an EO image. The algorithm, adapted from (Bower 2015), was slightly modified in order to enable it to detect this degradation. The work presented in the previous paper did not require a preprocessing step for the data. In this work, a data preprocessing step was found to be necessary in order to accomplish image blurring detection. The application of SA to this problem was chosen because 1) the algorithm could be used in a platform utilizing radar and EO systems for combined degradation detection and 2) the implementation of the algorithm does not overly impact the Size, Weight, and Power (SWaP) of the payload. The ability of the algorithm to run in real-time also allows for operation modification to the platform in order to mitigate any detected degradation. The objective of this work was to determine if the SA algorithm could be adapted in this sense.

Expanding upon this introduction, the paper is broken down into sections as follows. Section 2 reviews the SA algorithm and with Section 3 detailing the preprocessing and approach implemented in diagnosing image blurring. Section 4 investigates EO examples of blurring and the results from the algorithm. Section 5 concludes the paper with suggestions and future work.

2. SYMBOLIC ANALYSIS ALGORITHM

Expanding upon the work originally presented in (Bower, 2011), the objective was to design a system that was capable of integrating the developed software into the radar system. The software, Symbolic Analysis (SA), is briefly reviewed here.

2.1. Symbolic Analysis

Symbolic Analysis is a statistical pattern recognition tool based upon symbolic theory. Most work in the symbolic realm deals with the development of optimal models to determine the trajectory of modeled system states (Daw, Finney & Tracy, 2003). These methods are used to model complex and chaotic systems. The resultant optimal model, known as the ε machine (epsilon machine), has a variable dimensional structure whose dimensions were constantly adjusted depending on the data collected over time in order to maintain optimality. This variation in dimensionality made it difficult to determine deviations between models developed through system usage. In order to make meaningful comparisons between models, a machine was developed with a-priori fixed dimensional structure (Ray, 2004). This fixed dimensional machine allows for meaningful comparisons between statistical models defined at different temporal points in the system's life at the cost of model optimality. The process of SA is shown in the block diagram of Figure 1. Each process will be briefly described.

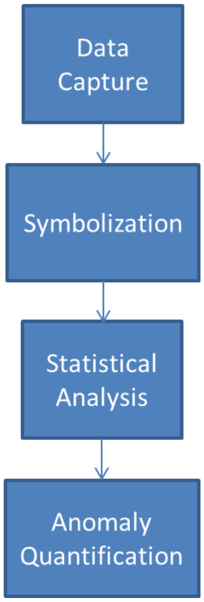

Figure 1. Symbolic analysis of time series data block diagram.

2.2. Data Capture

The data capture is an important step as it identifies data sources that are related to underlying degradation signatures. In addition to identifying relevant observables, the SA approach requires two assumptions: 1) the system does not undergo 'self-healing', and 2) that the underlying degradation dynamics can be separated from the system dynamics.

Assumption 1 forces the system to undergo a monotonically increasing degradation state, which assists in predicting future failure. Assumption 2 is far more stringent in that the data captures must be sufficiently long enough to develop statistics but also not capture changes in the underlying dynamics of the system. In other words, it is assumed that the degradation dynamics evolve at different rates then the system operates. For the EO platform, the image in of itself is a static representation of its environment and operating conditions. In this case, the operating conditions include the camera settings and the current conditions of the camera platform.

2.3. Symbolization

The next step involves transforming the image data into the symbolic domain. This step can be thought of as a general re-quantization of the original data resulting in a coarser distribution. Symbolization requires the determination of the number of partitions to be used as well as the type of partitioning. The two most common types of partitioning include uniform partitioning (UP) and maximum entropy (ME) partitioning. QorTek has devised a new partitioning approach which combines the advantages of both UP and ME which is called Mixed (MX) partitioning. MX partitioning was implemented in the analysis of image degradation.

2.3.1. Partitioning

That partitioning scheme of the algorithm allows for the collected data to be converted into the symbolic space. As stated previously, there are three main approaches utilized for partitioning including UP, ME, and MX although the majority of this write-up will focus on MX partitioning.

Uniform partitioning divides the range of the collected data into equal sized regions where the total number of determined partitions are defined as the set P. Given the range of the data as U, the partition sizes are defined as $^{U}/_{P}$ and the boundaries developed from the range U. Each partition region P_i was mutually exclusive and exhaustive over the range of the data. The probabilities of the partition occurrence in the uniform case are not necessarily equal; however, the partitioning structure was equal.

The maximum entropy (ME) partitioning scheme was defined by the principle of entropy in determining the partition structures. Recall entropy as described in Eq. 1.

$$H(X) = -\sum_{i=1}^{n} p(x_i) \log_2 p(x_i) \qquad (1)$$

The entropy can be maximized by setting $p(x_i) = p(x_j), \forall i, j$. The logarithm to base 2 was used so that the unit of entropy is in bits. In the data, accomplishing maximization of entropy in the baseline case was necessary to make sure all partitions (or symbols) have equal probability of occurrence. The partition structure resulting from ME does not necessitate equal partitions as in the uniform case but does guarantee equal prior probabilities for the partitions in the baseline case. A feature of the ME partitioning scheme is that the partitions boundaries are closer in regions of the data where there are a dense number of data points. In regions where there are fewer data points, fewer partitions are generated in these areas.

Mixed partitioning was developed at QorTek to combined the sensitivity of the ME approach with the equal area distribution of the UP approach. In the work completed, the MX approach takes the desired number of partitions and divides them equally between those to be developed through ME and those to be developed under UP. The resultant MX partitioning approach very finely models regions of dense data and uniformly divides other regions to allow for evolution of the system. This also enables the algorithm to model slight changes in the data-dense regions while allowing for more significant system deviations through the UP partitions.

Once the partitions are defined each partition was labeled with a symbol from the alphabet S. Given a data series X of length M, if $x_i \in P_i, 0 \leq i \leq M$, then assign $s_i \rightarrow x_i, \forall i;\ s_i \in S$. By implementing the partition structure and assigning a unique symbol to each time series data point, the end result was called the symbol stream. This is the re-quantized time series data that is now transformed into the symbolic domain.

2.4. Statistical Analysis

Once the partitions have been developed and symbols assigned to each partition, the next step is to construct the statistical model based on the resultant symbol stream. This step is controlled by another parameter for the SA methodology, the depth parameter D. The depth parameter controls the definition of model states from the symbol stream. States in the model are formed from D-length subsets of symbols. Therefore, the total number of states in the algorithm given the number of partitions P and the depth D is shown in Eq. (2).

$$N_s = P^D \qquad (2)$$

Equation (2) holds true independent of the partitioning scheme utilized. As an example, assume a ternary partition scheme is implemented that results in three symbols; labeling them -1, 0, and 1. The methodology's resultant statistical states depend on the number of symbols in the algorithm as well as the chosen depth. The parameter depth adjusts the memory of the resultant symbolic model, that is, the parameter controls the groupings of symbols into states. For instance, if D was unity, the resultant states are 0, 1, and -1. If D was two, the resultant states would be 00, 01, 10, 11, 0-1, (-1)0, (-1)(-1), 1(-1), and (-1)1. Therefore, the total number of states follows (2).

Shown in Figure 2 is an example model formation with the three-partition symbolic system and with D being equal to two applied to a recorded sine wave of arbitrary amplitude. These parameter choices result in a model with three states. The example sine wave in the figure is divided into zero (0), one (1) or minus one (-1) by a set threshold (partition boundary, uniform in this example). The symbol sequence is the square wave in the figure.

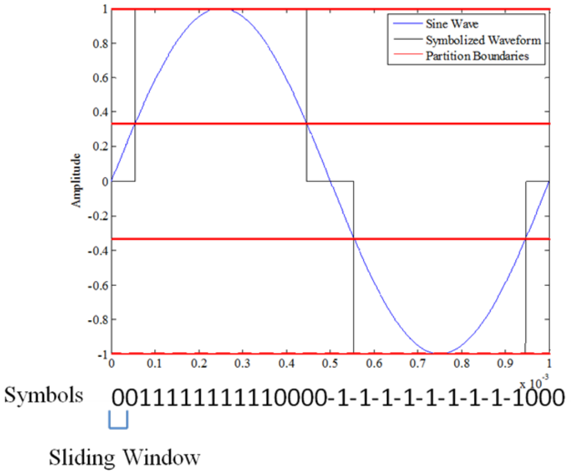

Figure 2. Example symbolization using three symbols with *d*=2 resulting in nine possible states.

With the symbol sequence s_i completed, the next step is to form states out of the symbols or groups of symbols. The probabilities of the state occurrences can be calculated and tracked across each data capture. Counting state occurrences can then be converted into probabilities to generate what is known as the State Probability Vector (SPV): the probabilities are arranged in a $N_s x 1$ vector, where N_s represents the total number of states in the algorithm given by Eq. (2). In the case where depth of the algorithm is equal to unity, as it is in most cases, the total number of states is equal to the number of symbols used. Choosing *D* equal to unity results in the smallest possible model for a given number of symbols, thereby reducing computational complexity of the approach. For this study in applying SA to image blurring analysis, the depth parameter is chosen as unity.

In addition to tracking the probability of the model states, the transition probabilities can also be calculated. The transition matrix captures the dynamics of the symbolic model and it is possible to calculate the SPV given the state transition matrix as shown in Eq. (3).

$$v_i \Pi = \lambda_i v_i \tag{3}$$

In Eq. (3), Π is the state transition matrix, λ_i is the ith eigenvalue equal to unity, and v_i is the left eigenvector of Π associated with the unity eigenvalue. This state transition matrix model of the underlying data is leveraged in this paper in order to determine the 'degradation' or blurring within the image.

Once the probabilities or counts are known, a metric can be applied to the statistical model, SPV or state transition matrix of which the latter is used in this paper, and a measure of the image degradation can be identified.

2.5. Anomaly Quantification

Anomalies inherent to degradation in the system can be generated from the use of the SPV between the data captures. The metric quantifies the deviation between the known baseline, commonly known as the healthy state of the system, and a future system state. A previously used measure to quantify an anomaly between captures is based on the Euclidean distance given in Eq. (3) for pulse *j*.

$$A_j = \sqrt[2]{\sum_{i=1}^{N_S} \left(z_{i,nominal} - z_{i,j}\right)^2} \tag{4}$$

In Eq. (4), $z_{i,nominal}$ is the nominal (baseline) SPV state z_i and $z_{i,j}$ is the corresponding SPV state at iteration *j*. Since our current work is focused on a single image or single snapshot in time, this measure is not quite applicable as was in our previous work (Bower, 2015). However, a new measure can be quantified based on the total number of state transitions on the main diagonal compared to those on the off diagonal.

The anomaly can be used as a diagnostic measure to determine the amount of degradation the system has incurred over its lifetime or to be used as a prognostic measure. If training data exists for the system, the anomaly measure can then be used in a prognostic application to predict the remaining useful life of the system. In this paper, the anomaly itself is a measurement of the estimated amount of blurring in the image.

In previous work with the SA algorithm, the algorithm was applied to multi-dimensional data (two-dimensional data). Although this approach would seem appropriate for images analysis, this is not the case. For image processing, it was not required to process the image as in a two-dimensional manner. The data from the image is processed in a one-dimensional manner utilizing the columns of an image and moving down through the rows reshaping the image data into a one-dimensional vector.

3. PRE-PROCESSING OF THE IMAGE

A critical aspect of the algorithm application is to pre-process the data before implementation into the SA algorithm. In terms of Figure 1, the pre-processing occurs before symbolization. The SA algorithm analyzes the data and measures the total amount of high frequency information contained in the image. An image that contains blurring would thus contain smaller amounts of high frequency information compared to an image that has minimal blurring.

In order to assist in the determination of the image's high frequency content, a high pass filter is applied to the image. The filtering approach implemented is called the Sobel operator and is given as (for the X direction):

$$G_X = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \qquad (5)$$

The Y dimension of the operator is simply given as $G_Y = G_X^T$, which is the transpose of the X operator. More explicitly, the operator can be broken down into simply as a product of an averaging and differentiation filter:

$$G_X = \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} \begin{bmatrix} -1 & 0 & 1 \end{bmatrix} \qquad (6)$$

An example of the operation is as follows using the image shown in Figure 3. The image is a picture of a cat (Alton) which can be used to demonstrate the frequency content. The image size is 2256x3380 pixels (7.6 megapixels). Notice that the image, in terms of color contains very little significant deviation which will results in lower high frequency content. This can be demonstrated with the Sobel operator.

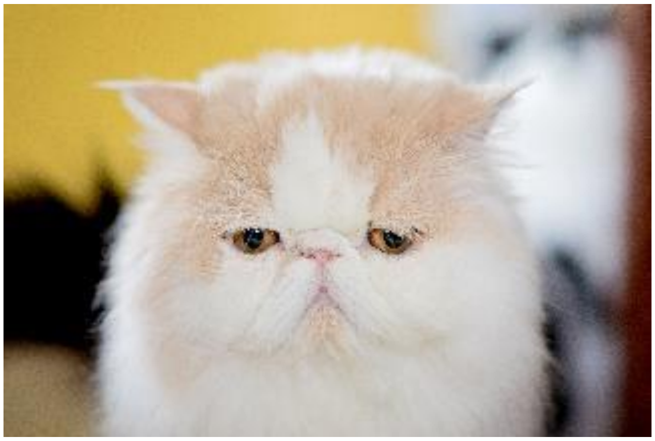

Figure 3. Example image, Alton's Monday face.

The first step with processing the image is to produce the magnitude image which is simple the red, green, and blue components averaged together into a single image. The grayscale image is shown in Figure 4.

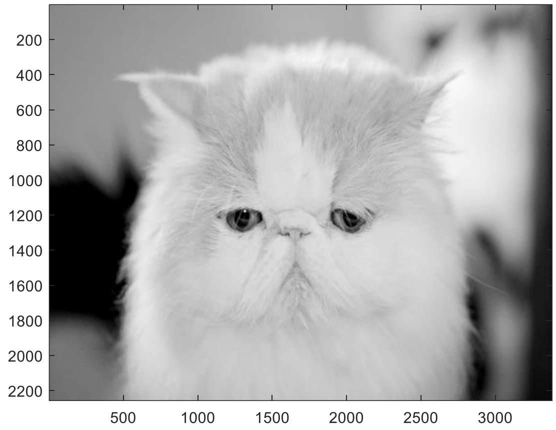

Figure 4. Grayscale image of example image.

The result of the Sobel operator applied to Figure 4 is shown in Figure 5. The figure shows regions where contrasts and edges are present which represent the high frequency information contained in the image.

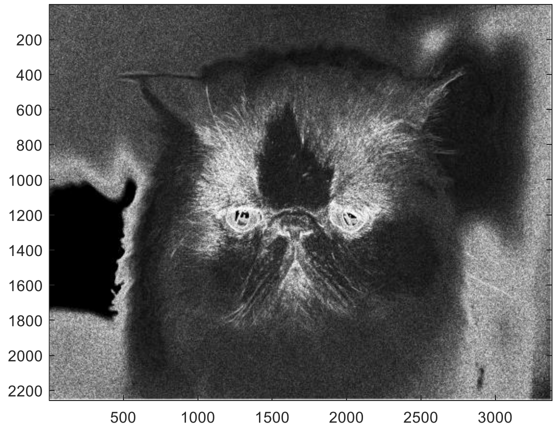

Figure 5. Gradient magnitude of example image.

The high frequency image information is thus utilized in the SA algorithm symbolization process. The objective would be to use the state transition matrix to determine the total amount of high frequency information. In essence, we are tracking the total number of occurrences of state transitions from one state to a state non-adjacent to the original state in the matrix. In other words, we are observing and tracking the total weight of transitions contained in the extreme off diagonal entries of the state transition matrix. This can be observed in the example matrix shown in Figure 6.

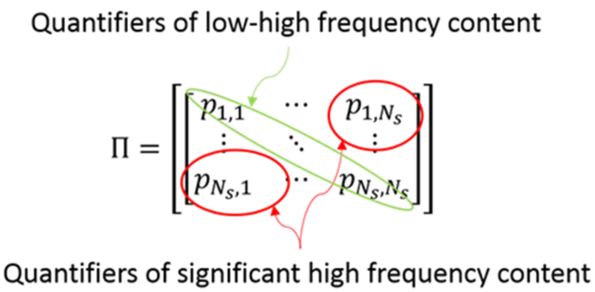
Quantifiers of low-high frequency content
$\Pi = \left[\begin{bmatrix} p_{1,1} & \cdots & p_{1,N_S} \\ \vdots & \ddots & \vdots \\ p_{N_S,1} & \cdots & p_{N_S,N_S} \end{bmatrix}\right]$
Quantifiers of significant high frequency content

Figure 6. Distribution of high frequency content in State Transition Matrix

It is shown that if the current state in the image is $p_{i,j}$ and the image transitions to $p_{i,j} \rightarrow p_{i+n,j+k}$, for i, k >1 where $0 \le i, j \le N_s$ and $n > 1$, that represents a transition to a non-adjacent state. N_s represents the total number of states formed in the state transition matrix. The key is observing states p_{1,N_s} and $p_{N_s,1}$ as well as states local to these extreme states. The higher the counts in these states indicate greater portions of high frequency information content in the image. Note that in this case, the symbolized image is mapped into a vector from a matrix for the statistical analysis. That is for an image of size M-by-N, the resultant vector size is thus MxN-by-1 and the statistical analysis carried out on this vector. To create the vectors, the columns N, are consecutively stacked to create the vector.

In general, in order to determine a threshold in which blurring is detected can be determined through training although an ad hoc method could be determined by thresholding the entire weight of the off diagonal entries versus the weight contained in the main diagonal. The general procedure is thus:

1. Filter the image under test utilizing a high pass 2D filter.
2. Utilizing the high frequency processed image, use this data in the symbolization portion of the SA algorithm.
3. Once the symbolization is complete, calculate the statistics of the 'symbolized image' using depth equal to unity.
4. Once the state transition matrix has been calculated, analyze the complete transitions contained in the extreme off diagonal entries and determine blurring probability,

Reiterating, step 4 above requires some intuition in determining a proper threshold for blurring detection. This process will be demonstrated utilizing the example image.

4. SA APPLIED TO BLURRING DETECTION

Utilizing the approach described in the previous section, the SA algorithm will be applied to the original (minimal blur) images and then applied to an artificially blurred image produced by Gaussian filtering of the original example image. We proceed first with the minimally blurred example.

4.1. Minimal Blur Example

In the previous section, the example image was processed and the image with the high frequency content remaining was shown in Figure 5. This image concluded the preprocessing step and the next step is symbolization of the high frequency image.

Before symbolization, the total number of partitions needed to be determined. Previous work has shown that the total number of pixels contained in the image can assist in determining the number of partitions. In order for the statistics of the state transition matrix to converge, there should to be enough samples per partition. In this work, it was experimentally found that if each partition contained at least 100,000 samples the algorithm results were positive.

Running the symbolization algorithm on the example image utilizing seven total partitions results in the image shown in Figure 7.

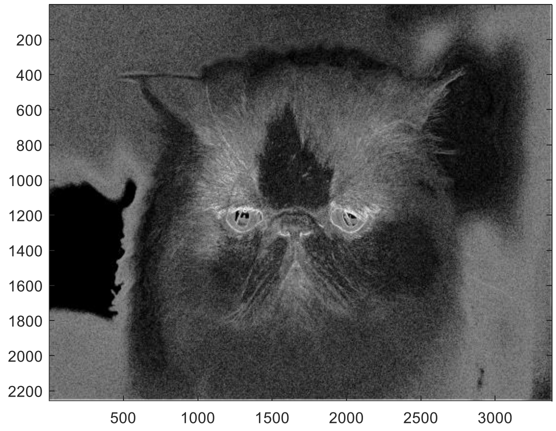

Figure 7. Symbolized results of the example image.

Comparing Figure 5 to Figure 7 shows the results between the raw high frequency image and symbolized image data. The total number of partitions used was adequate as the original image is still discernable meaning information loss is minimized between the original high frequency image and the symbolized image. This acts as a quick verification of the total number of partitions utilized

With the symbolization complete, the next step is to process the statistics of the symbolized image. Since the total number of partitions utilized in the example is seven and the depth is

Table 1. State transition matrix formed from minimally blurred example image.

State	p_0	p_1	p_2	p_3	p_4	p_5	p_6
p_0	835385	437883	246945	101073	1190	97	531
p_1	438418	590979	467638	227526	2804	225	754
p_2	246716	467327	614078	483993	7434	577	1115
p_3	100679	228135	483386	1343092	76192	5544	2777
p_4	1217	2863	7548	76100	30762	5454	2022
p_5	93	199	502	5596	5482	3583	2094
p_6	597	958	1143	2425	2102	2069	5898

unity, the total number of statistical model states is thus also seven. This means that the state transition matrix will be of $\mathbb{R}^{7x7}$.

The state transition matrix obtained from the original image is shown in Table 1. Note that the diagonal within the matrix is strong compared to the remaining off diagonal entries. This image presents an interesting aspect as there is not a significant amount of strong high frequency content. This can be observed in the original images (Figure 4) as there is not a significant amount of contrast in the magnitude image of the picture.

4.2. Blurred Image Example

With the original minimally blurred example shown, we now proceed to the purposely blurred images utilizing Gaussian filtering techniques (Trucco, 1998). The Gaussian isotropic two-dimensional filter is generally given as

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \qquad (7)$$

Utilizing this filter structure, Gaussian blurring filters of dimension 9 x 9 with varying sigma were generated. The filters used are summarized in Table 2.

Table 2. Gaussian Filter Parameters

Filter	Dimension	Sigma (σ)
#1	9x9	1
#2	9x9	2
#3	9x9	3

The Gaussian response is nearly infinite, but since after about three standard deviations the Gaussian response is nearly zero, the filter can be truncated. In this case, the truncation is approximately after nine sample points. The Gaussian filter response is shown in Figure 8 for filter #1. Note that with a small standard deviation, most of the filtering occurs within the nearest neighbor of image pixels. It is prudent to note that the filter had to complete dimension of 20 (as seen in the figure). The filter was also normalized in order to not affect the intensity of the filtered image.

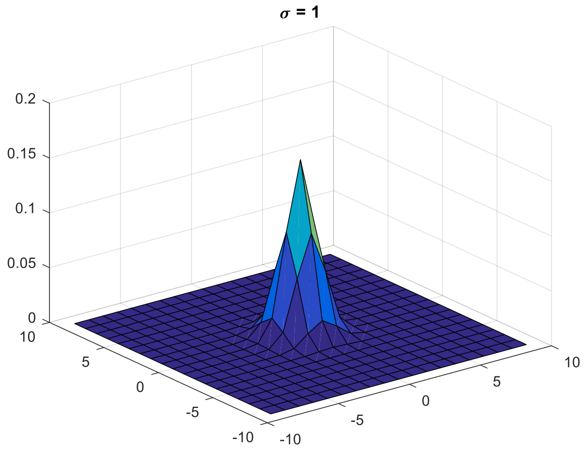

Figure 8. Gaussian filter response for filter #1.

Utilizing blurring filter #1, the image shown in Figure 9 is generated. Compare this image to that of the original image shown in Figure 4. There is some noticeable blurring around the edges of fur that can be seen. Pixel ranges of 1200 < y < 1800 and 400 < x < 600 is a good area to compare to observe this blurring. This range represents the edge of Alton's fur and comparisons between the two images shows an increase in blurring due to the filtering.

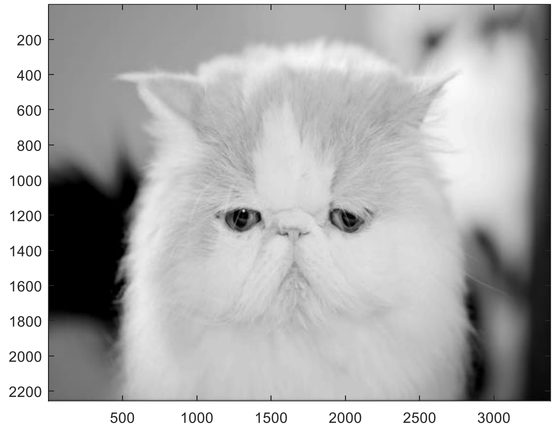

Figure 9. Blurred version of example image using filter #1.

To emphasize the increase in blurring, Figure 10 shows the Sobel filtered image. The reduction in high frequency lines can be concretely observed in this image. And finally, the resultant symbolized image of the Sobel filtered image is shown in Figure 11 which can be compared to Figure 7. The change between the minimally blurred image and the blurred image resulting from filter #1 is not as noticeable in the symbolized image.

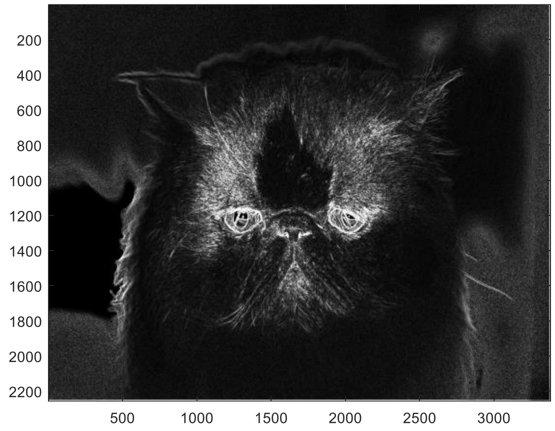

Figure 10. Sobel filtered, blurred example image utilizing filter #1.

The next set of figures increases the spread of the blur filter thereby increasing the amount of blurring in the image. The resultant image of filter #2 is shown in Figure 12.

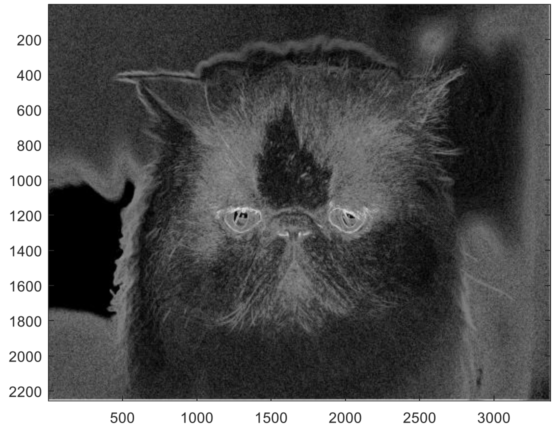

Figure 11. Symbolized image of filtered example image utilizing filter #1.

The Gaussian filter shown in the figure demonstrates the spreading of the filtering function from the increase in the sigma parameters. This will in turn increase the amount of blurring completed to the image which can be observed in Figure 13.

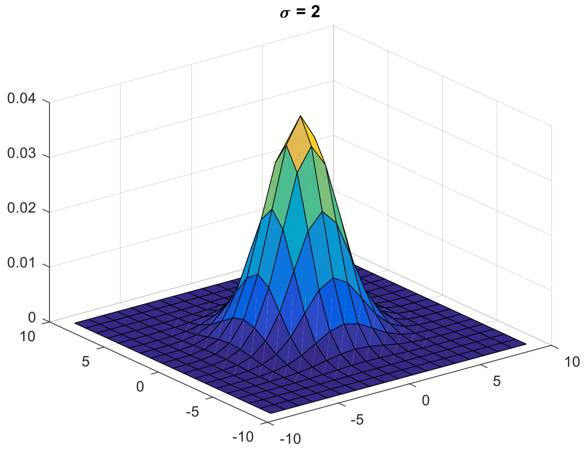

Figure 12. Gaussian filter response for filter #2.

Utilizing filter #2, the example image was filtered and the resultant blurred image is shown in Figure 13. Note that compared to the previous images (lesser filtered and original), facial details are more blurred in this result. To further emphasize the filtering effects, the Sobel filtered images is shown in Figure 14 which shows a decreased amount of high frequency content.

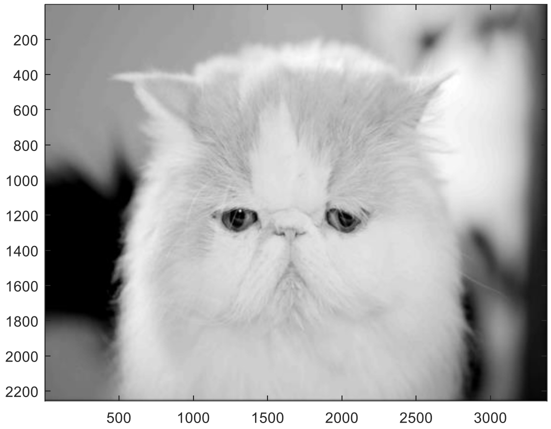

Figure 13. Blurred version of example image using filter #2.

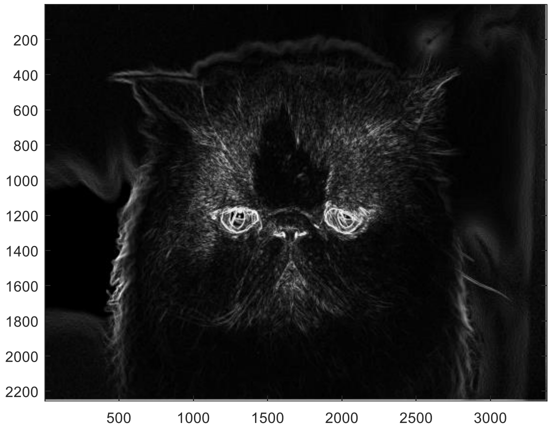

Figure 14. Sobel filtered, blurred example image utilizing filter #2.

The SA algorithm was applied to this Sobel filtered data and the resultant symbolic-based image is shown in Figure 15. Recall that a total of seven symbols were used in the SA algorithm. The image thereby scales these seven symbols evenly across the greyscale range of 256. It is visible in the image that there is a different probability spread of the symbols in the figure as observed in the background with reduced white areas. As the filtering becomes more intense, these white regions will continue to reduce because of blurring.

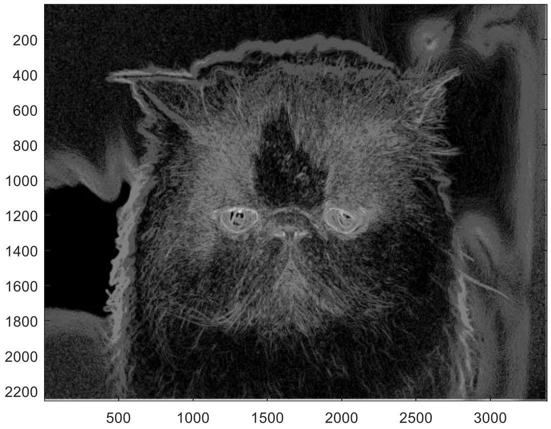

Figure 15. Symbolized image of filtered example image utilizing filter #2.

The next set of figures shows the results with utilizing filter #3. The filter's response is shown in Figure 16.

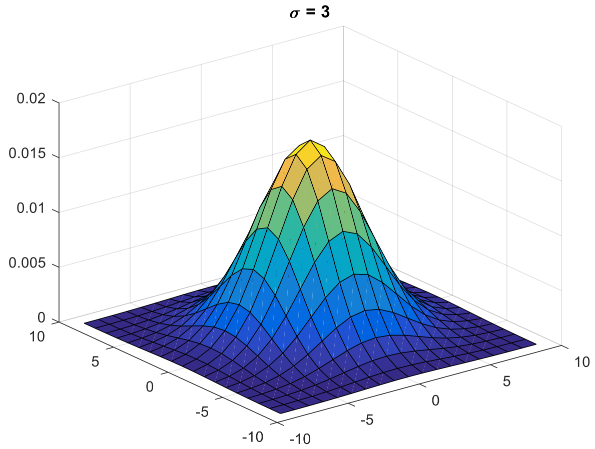

Figure 16. Gaussian filter response for filter #3

Note again the increase in the total number of pixels involved in the smoothing process. This in effect will further increase image blurring. Utilizing filter #3, a blurred version of the example image was again created. The blurred image is shown in Figure 17.

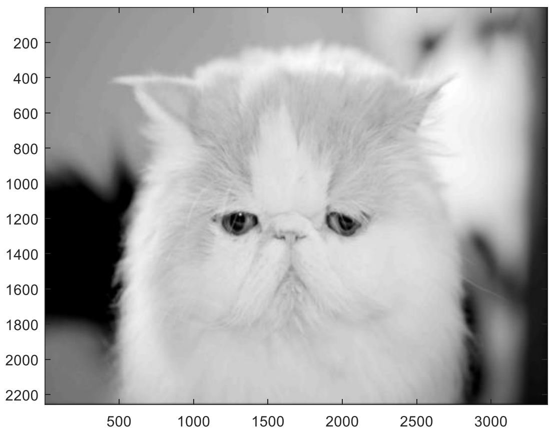

Figure 17. Blurred version of example image using filter #3.

As was completed previously, the image was high pass filtered in order to extract the high frequency information within of the image. The resultant image is shown in Figure 18.

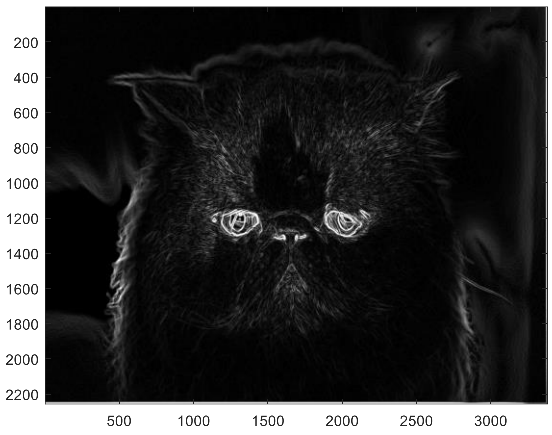

Figure 18. Sobel filtered, blurred example image utilizing filter #3.

Comparing Figure 18 with Figure 5 shows the amount of loss of high frequency content in the image due to the filtering operation. This Gaussian filter has removed a significant amount of the high frequency content.

.

The filtered image was thus used as the input into the SA algorithm and as previously, the symbolized image was also generated. This is shown in Figure 19.

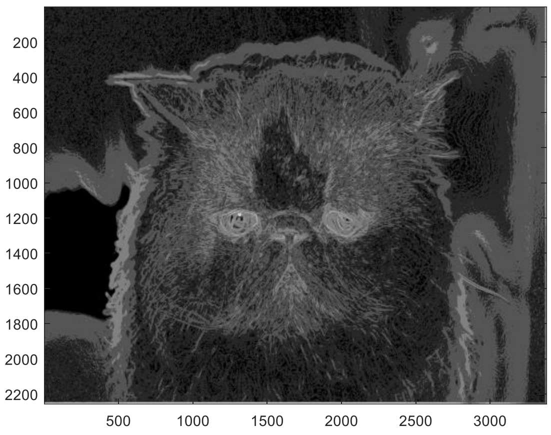

Figure 19. Symbolized image of filtered example image utilizing filter #3.

Comparing Figure 19 to the results of Figure 7 and the other symbolized images highlights the differences between them. The smoothing of the image creates a different probabilistic data set as compared to the original image and the state transition matrix is used to capture this difference.

4.3. State Transition Matrix Results

The means to deduce the amount of blurring in an image using the SA algorithm rests with analysis of the state transition matrix as observed in Table 1. By observing the total number of states transitions between the states, blurring can therefore be deduced. We now compare the results shown in Table 1, to the results obtained from blurring the image with filters #1, #2, and #3 which are shown in Table 3, Table 4, and Table 5 respectively. For all the tables below, the off diagonal entries are shaded.

The state transition matrices shown in Table 3, Table 4, and Table 5 show interesting results from the filtering operations. The different magnitude of blurring results in stark changes in the state transition matrix for each case. Comparisons of each of the above tables to Table 1 will highlight the changes to the state transition matrices that indicate blurring

Table 3. State transition matrix formed from filter #1 blurred example image.

State	p_0	p_1	p_2	p_3	p_4	p_5	p_6
p_0	929242	505161	126248	7317	17	1	601
p_1	504896	1164391	574705	30311	69	6	1252
p_2	126682	574370	1606389	283828	501	18	1216
p_3	7748	31638	285199	744786	13191	696	515
p_4	18	69	440	13377	14301	2124	472
p_5	1	1	15	2228	2200	4122	853
p_6	1	0	8	1926	522	2453	5075

Table 4. State transition matrix formed from filter #2 blurred example image.

State	p_0	p_1	p_2	p_3	p_4	p_5	p_6
p_0	1885161	332664	12885	444	1	4	1205
p_1	333474	1697836	228889	2013	0	4	1450
p_2	13281	231025	2149183	83996	18	2	691
p_3	448	2141	87226	473715	2797	48	19
p_4	1	0	13	5943	10527	828	16
p_5	0	0	0	282	2877	3060	129
p_6	0	0	0	1	1108	2402	3393

Table 5. State transition matrix formed from filter #3 blurred example image.

State	p_0	p_1	p_2	p_3	p_4	p_5	p_6
p_0	827131	148649	1035	4	0	3	349
p_1	148840	2917580	139263	104	0	13	2294
p_2	1197	141763	2745969	41962	3	1	696
p_3	4	102	45324	379467	1382	0	17
p_4	0	0	0	4732	14010	70	2
p_5	0	0	0	17	2825	1331	19
p_6	0	0	0	10	594	2774	1664

The first significant change is that a large region of the off diagonal matrix is less than that compared to the original matrix shown in Table 1. Specifically, the lower triangular portion of the matrix shows a decreasing amount of transitions. The upper triangular portion also shows a decreasing trend but not as significantly as the lower triangle portion of the matrix. A possible reason for more counts in the upper triangle region of the matrix could be due to the ordering of the columns of the matrix for the one-dimensional processing of the data.

With this shift in the distribution of transition probabilities, the state transition matrix of the filtered image has lost its weak symmetry along the main diagonal as transition counts are moved towards the diagonal. The filtered matrix still contains some transition counts in the upper triangle portion but these are significantly reduced as well.

Observing the changes in the other two example cases with the increased filtering further demonstrates the algorithm's application. As the amount of filtering increases, the total amount of counts in the off-diagonal entries further decreases. The results obtained from filter #3 shown in Table 5show a significant portion of zeroes in the off diagonal as well as the reduced counts in these entries.

From this example, it is easy to see how this approach can be used for image blur detection. A simplified analysis was carried out using the example image and this image was also significantly filtered to demonstrate the approach. A typical implementation could first analyze example images of terrain (urban, rural, water, etc.) in order to train the algorithm to determine a proper threshold of blur detection. For the example above, a significant loss of transitions in the off diagonal entries represented the loss of high frequency information.

Once these counts are calculated in the matrix, a threshold approach can be utilized in order to determine if blurring is present in an image. The determination of this threshold is key to the proper operation of the algorithm.

One means to generate a threshold is to sum all the transitions off of the main diagonal (shaded cells in the tables). This calculated value can also represent the amount of blurring in the image due the applied filtering.

Using this total sum of transitions, the percentage of these transitions is calculated over the entire state transition matrix. The figure of merit for Table 1 was calculated as 15.87%. The figures of merit for the other cases are shown in Table 6 as well as for the original image. These figures demonstrate or quantify loosely the total amount of blurring added to the image due to the Gaussian filtering. Additional figures of merit could be identified with additional research.

As can be seen in the figure, as the amount of filtering increases, the total number of counts in the off-diagonal entries (shaded) decreases significantly. The end results are that with filter #3, the total percentage of off-diagonal entries is 0.085% which is significantly less than the original 15.87% of the original image.

Table 6. Figures of merit for filtered cases

	Non-Shaded	**Shaded**	**Percentage**
Original	6,369,813	1,201,387	15.868%
Filter #1	7,230,663	340,537	4.497%
Filter #2	7,535,125	36,075	0.476%
Filter #3	7,564,755	6,445	0.085%

Also, note that as the total amount of blurring is increased by the filtering operation, the percentage of off diagonal entries further decreases. This behavior was expected as blurring increases, high frequency content decreases thereby decreasing the probability of a state transitioning to a non-adjacent state. This in turn reduces the total amount of off diagonal transitions as observed in the state transition matrices for each of the filtered results.

5. Conclusion

In this paper, we have demonstrated a means to identify blurring within an image without necessarily observing the image. The challenge with any blind approach is to appropriately train or learn patterns that will indicate the desired response without knowing the input to the system. In this case, a certain percentage of transition outliers in the state transition matrix can be used as the indicator of blurring in the image. The SA algorithm,

adapted from previous work, has shown promise in this regard and also can be implemented with a limited amount of available platform resources.

Future work would include further investigations on metrics that can be used to identify blurring in the image. This paper presented a straightforward approach simply counting the total number of off diagonal transitions and comparing that to the total transitions contained in the state transition matrix. Other metrics that could be implemented could focus on the strength of the main diagonal or even utilize the state probability vector.

From our previous work, we understand the difficulty that different terrain can have on data driven algorithms utilizing terrain-based data. Future work would investigate the effect terrain has on the results of the algorithm.

Finally, a simplistic method was presented on how to choose the total number of partitions for the algorithm. A more thorough analysis on partitioning type and perhaps depth may lead to improved results.

ACKNOWLEDGEMENT

This work was support by an AFRL contract FA8650-13-C-2318. QorTek would also like to thank Thierry Pamphile AFRL Wright-Patterson for his support and guidance during this work.

NOMENCLATURE

A	=	anomaly
D	=	symbolic depth
$H(\cdot)$	=	entropy
M	=	time series data length
N_s	=	number of states
$p(\cdot)$	=	probability
P_i	=	i^{th} partition
s_i	=	i^{th} symbol
U	=	time series data amplitude range
X	=	time series data
z	=	state probability vector
v_i	=	i^{th} eigenvector
λ_i	=	i^{th} eigenvalue
Π	=	state transition matrix

REFERENCES

Bower, G., Mayer, J., & Reichard, K. (2011). "Symbolic Dynamics and Analysis of Time Series Data for Diagnostics of a dc-dc Forward Converter," in *Annual Conference of the Prognostics and Health Management Society*, Montreal, 2011.

Bower, G., Zook, J., Bird, R. (2015). “Application of Health Management and Diagnostics for Synthetic Aperture Radar (SAR) Payloads” in *Annual Conference of the Prognostics and Health Management Society*, San Diego, 2015.

Daw, C.S., C.E.A. Finney & E.R. Tracy (2003). "A review of symbolic analysis of experimental data." Review of Scientific Instruments 74.2 (2003): 915-930.

K. Abed-Meraim, K., Qiu, Wanzhi., Hua, Yingbo (1997). "Blind system identification," in Proceedings of the IEEE, vol. 85, no. 8, pp. 1310-1322, Aug 1997.

Koik, B. T.; and Ibrahim, H. (2013) "A Literature Survey on Blur Detection Algorithms for Digital Imaging," Artificial Intelligence, Modelling and Simulation (AIMS), 2013 1st International Conference on, Kota Kinabalu, 2013, pp. 272-277.

Ray, Asok (2004). "Symbolic dynamic analysis of complex systems for anomaly detection." *Signal Processing* (2004): 1115-1130.

Tiwari, Shamik., Shukla, V. P. , Biradar, S. R., and Singh, A. K., (2014). “A Blind Blur Detection Scheme Using Statistical Features of Phase Congruency and Gradient Magnitude,” Advances in Electrical Engineering, vol. 2014.

Tong, Hanghang; Li, Mingjing; Zhang, Hongjiang; and Zhang, Changshui (2004). "Blur detection for digital images using wavelet transform," Multimedia and Expo, 2004. ICME '04. 2004 IEEE International Conference on, 2004, pp. 17-20 Vol.1.

Trucco, E., Verri, A., (1998). “Introductory Techniques for 3-D Computer Vison,” Upper Saddle River, NJ, Prentice Hall.

BIOGRAPHIES

Gregory Bower is currently the CTO of QorTek Inc. in Williamsport, PA. He received his B.S., M.S., and Ph.D. degrees in Electrical Engineering all from The Pennsylvania State University. Previously, he had worked as an assistant with the Applied Research Laboratory in State College, PA. His research interests include Prognostics and Health Management (PHM) of electronic systems, robust and optimal control theory, system identification, and power conversion.