Masked Object Registration in the Fourier Domain

Dirk Padfield
GE Global Research
padfield@research.ge.com

Abstract—Registration is one of the most common tasks of image analysis and computer vision applications. The requirements of most registration algorithms include large capture range and fast computation so that the algorithms are robust to different scenarios and can be computed in a reasonable amount of time. For these purposes, registration in the Fourier domain using normalized cross-correlation is well suited and has been extensively studied in the literature. Another common requirement is masking, which is necessary for applications where certain regions of the image that would adversely affect the registration result should be ignored. To address these requirements, we have derived a mathematical model that describes an exact form for embedding the masking step fully into the Fourier domain so that all steps of translation registration can be computed efficiently using Fast Fourier Transforms. We provide algorithms and implementation details that demonstrate the correctness of our derivations. We also demonstrate how this masked FFT registration approach can be applied to improve the Fourier-Mellin algorithm that calculates translation, rotation, and scale in the Fourier domain. We demonstrate the computational efficiency, advantages, and correctness of our algorithm on a number of images from real-world applications. Our framework enables fast, global, parameter-free registration of images with masked regions.

Index Terms—Registration, Masked Registration, Fast Fourier Transform, Normalized Cross-Correlation, Fourier-Mellin, Ultrasound, Tracking, Image Stabilization, Serial Sectioning

I. INTRODUCTION

Image registration is a fundamental task in image analysis whereby two or more images are aligned by finding a transformation that minimizes some distance between the transformed target image and the reference image. In this paper, our goal is to derive a mathematical framework for a registration approach that can exactly represent image masking fully in the Fourier domain using the Normalized Cross-Correlation (NCC) metric. Other researchers have solved parts of this problem. Some address the problem of registration using FFTs, others implement masking in the spatial domain, and still others incorporate masking into an iterative registration framework. There are also a number of efforts at registration using FFTs with other metrics, and a number of approaches focus on approximation methods that speed up the computations. We will demonstrate how to combine the strengths of these approaches and show that there is real practical value in an exact solution that represents the masking step fully in the Fourier domain.

Registration algorithms can be divided into a large number of categories [1], [2] including algorithms that use image pixel values directly [3], algorithms that operate in the Fourier domain [4], [5], [6], [7], [8], [9], and algorithms that register based on image features [10], [11], [12]. Furthermore, registration algorithms can be either rigid or deformable. In this paper, we are concerned with addressing rigid transformations as these are useful in themselves for a large range of applications and are also commonly used as the initialization step of deformable registration approaches. In particular, we are concerned with the class of registration algorithms computed in the Fourier domain, which is of particular interest for a number of reasons. First, only one iteration of the algorithm is needed to provide a global registration result that calculates the metric for all possible transforms between the images. This is important in applications where the images are significantly misaligned so that few constraints can be placed on the transformation between them. This is in contrast to algorithms that start from a given transform, test the metric after moving in a particular direction, and iterate this process. Such algorithms only calculate the metric for a small region of the overlap, can become stuck in local optima, and can take a long time to converge. Second, Fourier domain algorithms are fast because they require only a small number of forward and backward transforms along with simple mathematical operations. Third, such approaches require few or no parameters.

This paper focuses on the use of the correlation metric since the cross-correlation of two signals has a dual in the Fourier domain. Correlation is the most widely used method for similarity detection [3] and the automatic determination of translation [13], [14], and in [15] Lewis stated that, “it is probably fair to say that a suitable replacement has not been universally recognized”. The use of cross-correlation for image alignment is motivated by the squared Euclidean distance. However, cross-correlation by itself is not invariant to changes in intensity amplitude and features size. This is also true for other commonly used metrics like sum of squared differences (SSD). The normalized cross-correlation (NCC) overcomes these difficulties by normalizing the image and feature vectors to unit length, yielding a cosine-like correlation coefficient. For images from the same imaging modality, the NCC metric is appropriate because it is insensitive to multiplicative factors between the two images and produces a cost function with sharp peaks and well-defined minima. The approach of Lewis in [15] is one of the seminal approaches for fast template matching using NCC. The popularity of the approach results from its use of integral images for efficient computation of the normalization factors. Briechle and Hanebeck in [16] build upon this fast method for template matching by using
basis functions for approximating the templates. While the generation of the basis functions for arbitrary templates is not trivial, the approach is very effective and fast if appropriate basis functions can be generated. In addition to the NCC metric, the use of phase correlation with FFTs for registration has also been extensively studied [17], [18].

Although FFT NCC methods are extensively used for translation registration and template matching, they have also been extended to rotational transforms [5], and further extended to also include scale [9], [19], [6], [7], [8], which enables similarity transforms calculated entirely in the Fourier domain. This is accomplished by making use of the translation, rotation, and scale properties of Fourier transforms and their duals in the spatial domain. In particular, the displacements between images do not affect the magnitudes of the Fourier transforms, spatial rotation is the same as rotation in the frequency domain, and scaling in the spatial domain corresponding with an exponential scale in the frequency domain. Thus, the conversion of the magnitude of the Fourier transforms of the images to log-polar coordinates converts the scale and rotation differences to vertical and horizontal offsets that can be measured. Using these properties, these algorithms operate by first computing the scale and orientation by calculating the correlation of the magnitude of the Fourier transforms, transforming the original images using the calculated scale and orientation, and then correlating the transformed images to compute the translation component. This approach is widely used and successful but is generally limited to a scale change of a factor of 2.

Building upon this initial concept of Fourier domain registration for similarity transforms (translation, rotation, and scale), a large number of approaches have been developed to improve the accuracy for more robustness to large scale and rotation changes. In [20], Keller et al. introduce an approach for more accurate polar and log-polar processing of the Fourier transform magnitudes. Rather than the standard approach of taking the FFT on the Cartesian grid and then interpolating on the polar or log-polar grid, this work directly transforms image into polar Fourier domain by using the separable pseudo-polar FFT in different dimensions. The approach was demonstrated to handle scales up to 4 on images of objects cropped from the background. However, it still requires interpolation, fails to accurately recover rotations, and is iterative unlike the original Fourier domain algorithm. To address some of the limitations of the pseudo-polar approach, Pan et al. in [21] introduce a multi-layer approach for generating an FFT grid adapted to the polar and log-polar transform. They demonstrate improved results and an ability to handle scales up to 10 with no rotation and 5 with rotation. The approach requires setting the parameters of the number of layers and the optimization of the approximation set for each set of layers.

While the approaches of [20], [21] focus on the Fourier domain interpolation grid, the orientation correlation method of Fitch et al. in [22] utilizes the image gradient instead of the intensity image by embedding the two components of the gradient into a complex image for correlation registration. Using the gradient makes the correlation independent of intensity scaling and offset and can be adapted to images of different modalities. However, the method is limited to translation registration. This method is generalized to rotation and scale by Tzimopoulos et al. in [23], which also uses intensity gradients. This approach introduces a normalized gradient correlation approach which is able to capture a much greater range of orientation and scale changes than using image intensities because of the ability of the gradients to more accurately capture salient image information.

To enable full deformable registration, the approach of Larrey-Ruiz et al. in [24] embeds the variational minimization of a number of energy functional directly into the Fourier domain. This enables a speed-up of twice over variational minimization in the spatial domain.

Rather than operating in the Fourier domain, some approaches compute the rotation and scale in the spatial domain. Such approaches solve for the rotation and scale by choosing the center of rotation to be the same in both images, but this approach suffers from the fact that it is difficult to find the correct center of rotation. Zokai and Wolberg in [25], [26] propose one solution to address this by using an exhaustive search of transformed cropped regions. All of the approaches discussed so far assume that the full images are to be registered and do not enable the specification of image masks.

During registration, it may be desirable to restrict evaluation of the metric within a specified region so that undesirable regions do not contribute to the metric calculation. For example, in the registration of ultrasound images (see Figure 1), the background region surrounding the "pie-slice" containing the actual image information should be ignored. If the pixels in such regions contribute to the metric calculation, this can lead to incorrect registration results. Various approaches have been proposed for such masking. Kaneko et al. in [27] propose an algorithm that incorporates a binary mask into the NCC equation, but, in the process of masking the images before correlation, the masked regions still affect the registration by lowering the sums used for normalization. This problem also exists for the approach of Lei in [28] where the images are masked by multiplying them with the binary masks. Alternatively, approaches such as those by Thevenaz et al. [29], [30] provide a mask under which the registration metric is to be calculated, but this masking is applied in an iterative framework, so a global solution is not calculated. Furthermore, approaches that calculate the masked NCC in the spatial domain for all locations become computationally infeasible for large images. Stone et al. in [31], [32] present a masking step based on correlations for the application of measuring differences between images and patterns.

Fitch et al. in [33] introduce a robust FFT correlation approach.
method for translational transforms that is robust to outliers by approximating kernels with a number of sinusoids. The correlation is derived based on an error surface, which also incorporates masking in the form of “alpha masks”. The method yields an approximate registration whose results are controlled by parameters that enable a trade-off between accuracy and computational complexity. An approach for using FFTs with the sum of squared differences (SSD) metric to register multimodality image pairs that include regions of interest (ROIs) is presented by Orchard in [34]. In this approach, a linear remapping step is incorporated into the equation for the SSD that enables it to adapt to modalities that have different intensity distributions. The algorithm results in an integer pixel-shift solution for translation (with minor rotations of up to 5°) that can be followed by a local search to find the exact alignment. An ROI can be specified, but it is limited to being entirely inside of the other image in the shifting process.

Table I summarizes the advantages and disadvantages of various approaches for the task of calculating global transformations in a computationally efficient manner using masked regions. It is desirable to integrate the advantages of these approaches into a consistent framework.

In order to meet all of the requirements of Table I to design an algorithm that can quickly find the global transformation of images that may be significantly misaligned and include masked regions, we present a method to integrate the masking directly into the FFT registration framework, which builds upon the approach we introduced in [35]. Our goal is neither to develop a new metric nor advocate the NCC metric over others. Instead, we introduce in this paper our approach including object tracking, image stabilization, and a number of previous applications of our algorithm covering the global search, speed, and parameter-free advantages of the Fourier transform and the appropriateness of the NCC metric with the power of masking undesirable regions in the images.

II. METHODS

In this section, we demonstrate how to derive mathematically the masked registration algorithm in the Fourier domain. First, in Section II-A, we give the standard form of normalized cross-correlation (NCC) of two images and show a form of this equation consisting exclusively of sums over the images. Although these sums can be efficiently calculated using a type of integral images, such methods will not work for the application of masked registration. However, we demonstrate that the sums can be represented fully in the Fourier domain and that this representation generalizes to the case of masked registration. Therefore, in Section II-B, we transform all of these sums into forms that are represented fully using FFTs. We show in Section II-C how this representation leads to a simple and general form for masked FFT NCC. Finally, in Section II-D, we demonstrate how the masked framework can overcome several limitations of the Fourier-Mellin method.

A. Spatial Form of Normalized Cross-Correlation

The 2D normalized cross-correlation indicates the similarity of two images $f_1(x,y)$ and $f_2(x-u,y-v)$, with $f_2$ shifted by $(u,v)$. Using this notation, we define that $f_1$ is the fixed image and $f_2$ is the moving image. In the process of shifting, we use $D_{u,v} = D_2(u,v) \cap D_1$ to represent the region of overlap of the two images, where $D_1$ is the domain of $f_1$ and $D_2(u,v) = \{x,y| (x-u,y-v) \in D_2\}$ is the domain of $f_2$ shifted by $(u,v)$. The region of overlap is constantly shifting and represents the overlapping region of the correlation operation. Furthermore, we define the mean intensity of $f_1$ and $f_2$ in the overlap region as

$$f_{1,u,v} = \sum_{(x,y) \in D_{u,v}} f_1(x,y)$$

and

$$f_{2,u,v} = \sum_{(x,y) \in D_{u,v}} f_2(x-u,y-v).$$

Using this notation, the normalized cross-correlation between images $f_1$ and $f_2$ at a given $(u,v)$ is defined as

$$\frac{\sum [(f_1(x,y) - f_{1,u,v}) (f_2(x-u,y-v) - f_{2,u,v})]}{\sqrt{\sum (f_1(x,y) - f_{1,u,v})^2} \sqrt{\sum (f_2(x-u,y-v) - f_{2,u,v})^2}}.$$
where all the sums are over \((x, y) \in D_{a,v}\). By expanding and combining terms, this can be rewritten as

\[
\text{NCC}(u, v) = \frac{\text{NCC}_{\text{num}}(u, v)}{\text{NCC}_{\text{den},1}(u, v) \sqrt{\text{NCC}_{\text{den},2}(u, v)}}
\]

(4)

where the numerator \(\text{NCC}_{\text{num}}(u, v)\) can be defined as

\[
\sum_{(x,y) \in D_{a,v}} f_1(x, y) f_2(x - u, y - v)
\]

\[
- \sum_{(x,y) \in D_{a,v}} f_1(x, y) \sum_{(x,y) \in D_{a,v}} f_2(x - u, y - v)
\]

(5)

and the \(\text{NCC}_{\text{den},1}(u, v)\) term can be represented as

\[
\sum_{(x,y) \in D_{a,v}} (f_1(x, y))^2 - \sum_{(x,y) \in D_{a,v}} 1
\]

(6)

and the \(\text{NCC}_{\text{den},2}(u, v)\) term is represented as

\[
\sum_{(x,y) \in D_{a,v}} (f_2(x - u, y - v))^2 - \sum_{(x,y) \in D_{a,v}} 1
\]

(7)

In Equation 5, the first term is simply the definition of the cross-correlation of the images. The numerator of the second term consists of the product of the local sum of \(f_1\) and the local sum of \(f_2\), and the denominator is the number of pixels in the overlapping region. The sums in the terms of Equations 5, 6, and 7 can be computationally intensive because they must be computed for each region of overlap of the two images. This amounts to \((q_1 + q_2 - 1) \times (r_1 + r_2 - 1)\) different sums, where \((q_j, r_j)\) are the rows and columns, respectively, of the images, and \(j = 1, 2\). Although there are algorithms to compute these sums more efficiently by precomputing running sums such as integral images, these algorithms will have difficulty when using masked regions. We therefore seek to represent all of these sums directly in the Fourier domain. The motivation is that the computation is much faster in the Fourier domain: spatial correlation has an order complexity of \(O(n^2)\) whereas Fourier correlation has an order complexity of \(O(n \log(n))\), where \(n\) is the number of pixels in the image regardless of image dimension. This difference in computation time becomes more significant with increasing image size.

\[B. \text{ FFT Form of Normalized Cross-Correlation}\]

Our next goal is to define the entire normalized cross-correlation operation in the Fourier domain for the standard NCC without masking. We will then show in Section II-C that this representation enables a simple definition of masked NCC.

Representing all terms of NCC in the Fourier domain would not only lead to faster processing, but it would also enable a compact mathematical representation of the computation. In [15], Lewis sought to accomplish this but came to the conclusion that NCC “does not have a simple frequency domain expression.” Therefore, his algorithm was “computed in the spatial domain for this reason” using integral images to represent the terms in the denominator. In this section, we show that NCC actually does “have a simple frequency domain expression.”

It is well known that the spatial form of correlation has a simple dual in the Fourier domain [36]

\[
\int_{-\infty}^{\infty} f_1(\tau) f_2(\tau - \tau)d\tau = \mathcal{F}^{-1}(\mathcal{F}(f_1) \cdot \mathcal{F}(f_2))
\]

(8)

where \(f_2'\) is \(f_2\) flipped (or, in 2D, rotated by 180°). For simplicity of notation, we will therefore carry out all operations on this rotated image. We will also define \(F_1 = \mathcal{F}(f_1)\), and \(F_2' = \mathcal{F}(f_2)\), where \(\mathcal{F}(\cdot)\) represents the FFT operation and \(F'\) is the complex conjugate of the Fourier transform, which, by definition, is the Fourier transform of the rotated image for real-valued images. Also, assuming \(i_1\) and \(i_2\) are images of the same size as \(f_1\) and \(f_2\), respectively, we define \(I_1 = \mathcal{F}(i_1)\) and \(I_2' = \mathcal{F}(i_2)\). Furthermore, the size of each dimension of all of the FFT images in the following equations is set to \(\text{max}(q_1 + q_2 - 1, r_1 + r_2 - 1)\), where \((q_j, r_j)\) represents the number of rows and columns of the images, and the subscripts refer to the two images. This is accomplished by padding the images with zeros before calculating the FFT. This zero-padding also avoids the problems of periodic boundary conditions in the FFT computation. Note that all of the FFTs could be different from each other.

The first term of Equation 5 is the cross-correlation of the two images, so it can be calculated in the Fourier domain as \(\text{CC}(f_1, f_2) = \mathcal{F}^{-1}(F_1 \cdot F_2')\). To accomplish the conversion of Equation 4 to the Fourier domain, we need to re-define the following local sums in terms of Fourier transforms as follows

\[
\sum_{(x,y) \in D_{a,v}} 1 = \mathcal{F}^{-1}(I_1 \cdot I_2')(u, v)
\]

(9)

\[
\sum_{(x,y) \in D_{a,v}} f_1(x, y) = \mathcal{F}^{-1}(F_1 \cdot I_2')(u, v)
\]

(10)

\[
\sum_{(x,y) \in D_{a,v}} (f_1(x, y))^2 = \mathcal{F}^{-1}(\mathcal{F}(f_1) \cdot F_2')(u, v)
\]

(11)

The rest of the terms in Equations 5, 6, and 7 can be converted analogously.

The local sum in Equation 9 calculates the number of pixels in the overlap region for a particular value of \((u, v)\). Because the correlation operation by definition calculates the sum of the product in the overlap region as the moving image is shifted across the fixed image, we can define this sum in Equation 9 as the correlation of two images whose values are all 1. The sum in Equation 10 is the local sum of image \(f_1\), which can similarly be represented as the correlation of this image with an image of ones in Equation 10. Similarly, the sum in Equation 11 is the local sum of the element-wise square of
image \( f_1 \), which can be represented by the correlation of this image with an image of ones.

Given these conversions, Equation 5 becomes

\[
\text{NCC}_{\text{num}} = \mathcal{F}^{-1}(F_1 \cdot F_2^*) - \frac{\mathcal{F}^{-1}(F_1 \cdot F_2^*) \cdot \mathcal{F}^{-1}(I_1 \cdot F_2^*)}{\mathcal{F}^{-1}(I_1 \cdot F_2^*)},
\]

Equation 6 becomes

\[
\text{NCC}_{\text{den},1} = \mathcal{F}^{-1}(\mathcal{F}(f_1 \cdot f_1) \cdot I_2) - \left(\frac{\mathcal{F}^{-1}(F_1 \cdot F_2^*)}{\mathcal{F}^{-1}(I_1 \cdot F_2^*)}\right)^2,
\]

and Equation 7 becomes

\[
\text{NCC}_{\text{den},2} = \mathcal{F}^{-1}(I_1 \cdot \mathcal{F}(f_2' \cdot f_2')) - \left(\frac{\mathcal{F}^{-1}(I_1 \cdot F_2^*)}{\mathcal{F}^{-1}(I_1 \cdot F_2^*)}\right)^2.
\]

Note that, whereas Equations 4, 5, and 6 are defined for a particular value of \((u, v)\), Equations 12, 13, and 14 are defined for all values of \((u, v)\). All of the terms are matrices, and all of the multiplications and divisions in these equations are element-wise.

This construction requires the calculation of 6 forward FFTs (only 5 are required if the image size of \( f_1 \) and \( f_2 \) are the same since \( I_1 \) and \( I_2 \) will then be equal) and 6 backward FFTs. However, since the 6 forward FFTs all have as their input real images, we embed them in pairs into complex images, calculate 3 FFTs, and then use odd-even separation to extract their FFTs, thus saving 3 FFT computations. These forward and backward FFTs make up the majority of the computational complexity.

C. Masked FFT NCC

In this section, we extend the FFT NCC formulation to enable the correlation of images that have associated masks. The beauty of representing the NCC completely in the Fourier domain in Section II-B is that its generalization to the masked FFT NCC becomes straightforward. We will demonstrate how their FFTs, thus saving 3 FFT computations. These forward and backward FFTs make up the majority of the computational complexity.

Assume \( m_1 \) and \( m_2 \) are mask images of the same size as \( f_1 \) and \( f_2 \), respectively. In these masks, regions of interest are given by a value of 1, and regions to be ignored (masked out) are given a value of 0. To account for the masked regions, in the process of shifting, we use \( D_{u,v} = D_{2,m}(u,v) \cap D_{1,m} \) to represent the region of overlap of the two images where neither of the masked images are 0. Here \( D_{1,m} \) is the domain of \( f_1 \) in the areas that \( m_1 \neq 0 \), and \( D_{2,m}(u,v) = \{x,y|(x-u,y-v) \in D_{2,m}\} \) is the domain of \( f_2 \) shifted by \((u,v)\) in the areas that \( m_2 \neq 0 \). Note that this notation ensures that the regions in the masks that are set to 0 have no influence in the overlap region. This is in contrast to simply masking the fixed and moving images and then correlating them using Equation 4, which would result in the zero values influencing the calculation. The difference is subtle but crucial. If the images are simply masked and then correlated, all pixels in the overlap region will be used in the computation of the correlation, including all of the zero values. However, what is required is that those values that are masked out have no influence in the computation and that the computation be carried out using only the valid regions. Figures 2 and 7 demonstrate visually the fundamental difference of these two approaches.

Using the domain \( D_{u,v,m} \), we can re-define the sums from Equations 9, 10, and 11. Figure 2 represents the equations visually by showing illustrations of the masked areas of the overlap regions. As in the case of correlating the images with images of ones in Equations 12 and 13, we define \( M_1 = \mathcal{F}(m_1) \) and \( M_2' = \mathcal{F}(m_2') \), where \( m_2' \) is \( m_2 \) rotated by \( 180^\circ \). To ensure that the masked out regions of \( f_1 \) and \( f_2 \) do not contribute to the sums, we re-define these images to be the point-wise multiplication with their corresponding masks \( f_1 = f_1 \cdot m_1 \) and \( f_2 = f_2 \cdot m_2 \) and calculate the Fourier transforms accordingly. Then the sums can be converted to FFTs as follows

\[
\sum_{(x,y) \in D_{u,v,m}} 1 = \mathcal{F}^{-1}(M_1 \cdot M_2')(u,v)
\]

\[
\sum_{(x,y) \in D_{u,v,m}} f_1(x,y) = \mathcal{F}^{-1}(F_1 \cdot M_2')(u,v)
\]

\[
\sum_{(x,y) \in D_{u,v,m}} (f_1(x,y))^2 = \mathcal{F}^{-1}((f_1 \cdot f_1) \cdot M_2')(u,v)
\]

Given these sums, the masked version of Equation 12 becomes

\[
\text{NCC}_{\text{num}} = \mathcal{F}^{-1}(F_1 \cdot F_2^*) - \frac{\mathcal{F}^{-1}(F_1 \cdot M_2') \cdot \mathcal{F}^{-1}(M_1 \cdot F_2^*)}{\mathcal{F}^{-1}(M_1 \cdot M_2')},
\]

Equation 13 becomes

\[
\text{NCC}_{\text{den},1} = \mathcal{F}^{-1}((f_1 \cdot f_1) \cdot M_2') - \left(\frac{\mathcal{F}^{-1}(F_1 \cdot M_2')}{\mathcal{F}^{-1}(M_1 \cdot M_2')}\right)^2,
\]

and Equation 14 becomes

\[
\text{NCC}_{\text{den},2} = \mathcal{F}^{-1}(M_1 \cdot (f_2' \cdot f_2')) - \left(\frac{\mathcal{F}^{-1}(M_1 \cdot F_2^*)}{\mathcal{F}^{-1}(M_1 \cdot M_2')}\right)^2.
\]
The advantages of representing the NCC fully in the FFT domain are clear from Equations 12, 13, and 14: it enables the representation of the masked NCC without the limitations of the standard NCC. The computation complexity for the masked algorithm is exactly the same as the complexity for the standard NCC. Thus, the masked NCC can be represented in the Fourier domain as in Equation 21. Note that, although we have used binary masks defined by 0 and 1 values, our derivations are general so that weighted masks defined by real numbers can be used. As an example, we can zero out the border regions that do not overlap with the original image to find the translation parameters. This image corrected for rotation and scale can then be correlated with the original image to find the translation parameters.

D. Masked FFT NCC for Translation, Rotation, and Scale Registration

The methods presented thus far have been concerned primarily with masked FFT NCC, which enables translation registration. However, they can also be applied to improve translation, rotation, and scale transforms using a framework similar to that of Reddy and Chatterji [9] and Lucchese et al. [37]. The method, commonly called the Fourier-Mellin method, is founded on the principles that the Fourier transform is translation invariant and that its conversion to log-polar coordinates converts the scale and rotation differences to vertical and horizontal offsets.

The Fourier-Mellin approach can be described as follows. Let $f_1(x,y)$ be the fixed image and $f_2(x,y)$ be the moving image. If $f_1(x,y)$ is a translated, rotated, and scaled version of $f_2(x,y)$ with translation $(x_0,y_0)$, rotation $\theta$, and scale $\tau$ (same in both dimensions) then the relationship is as follows:

$$
\begin{align*}
|f_1(x,y)| &= f_2(\tau \cos \theta + \tau y \sin \theta - x_0, \\
&\quad -\tau x \sin \theta + \tau y \cos \theta - y_0) \\
&= f_2(x,y)
\end{align*}
$$

The corresponding Fourier transforms of this relationship is

$$
\begin{align*}
F_1(u,v) &= \frac{1}{\tau^2} e^{-j2\pi(ux_0+vy_0)} F_2(u/\tau \cos \theta + v/\tau \sin \theta, \\
&\quad -u/\tau \sin \theta + v/\tau \cos \theta) \\
&= \frac{1}{\tau^2} N_2(u/\tau \cos \theta + v/\tau \sin \theta, \\
&\quad -u/\tau \sin \theta + v/\tau \cos \theta)
\end{align*}
$$

Thus, the magnitudes of both spectra are the same but are scaled and rotated versions of each other.

Transforming Equation 24 to polar coordinates, we find that

$$
N_1(\rho, \theta) = N_2(\rho/\tau, \theta - \theta_0)
$$

Furthermore, by converting the axes to logarithmic scale [38], scaling can be reduced to a translational movement as follows

$$
N_1(\log \rho, \theta) = N_2(\log \rho - \log \tau, \theta - \theta_0)
$$

The rotation angle and scale can then be found by correlating $N_1$ and $N_2$. The moving image $f_2(x,y)$ can then be corrected for rotation and scale by transforming the original moving image to $f_2(x,y)$. This leads to $f_1(x,y) = f_2(x-x_0,y-y_0)$. This image corrected for rotation and scale can then be correlated with the original image to find the translation parameters.

This algorithm, although widely used, suffers from two important limitations that can be overcome by using masked FFT NCC instead of the standard FFT registration. The first limitation is that when the moving image is rotated and scaled,
the background values must be set to some value, normally 0 (see Figure 3). These 0 values then have an effect on the correlation step and can lead to errors. Previous methods had no satisfactory way of handling these invalid regions: either they were included thus adversely affecting the registration result, or a smaller rectangular region was used that did not include the valid region, thus losing useful real information in the image. However, using masked FFT registration, those invalid regions can be entirely ignored by defining masks that are set to 1 inside the valid regions and 0 otherwise. This advantage is highlighted in Figure 3, and several examples on real images are provided in Figure 12.

The second advantage is that masking can be used in the log-polar transformed Fourier magnitude images. When a Cartesian image is transformed into the polar or log-polar domain, there are always curved regions of invalid pixels at distances that are farther than half the image width away from the image center (see Figure 4). Thus the image is generally sampled in the largest circular region that fits in the square image, but this ignores all of the valid pixels outside of this circle. Using masked FFT registration solves this problem by simply providing a mask for the invalid regions. Thus, even if masking of the original images is not used, this algorithm will be more accurate than that described in [9].

It is important to note that both of these advantages of using masked FFT registration are relevant even when the original images are not masked. Thus any applications using the standard Fourier-Mellin approach can be directly replaced with our algorithm. Our full algorithm for finding the transforms using masked registration is given in Algorithm 2. The algorithm lists two separate masked FFT registration steps. In the first, the inputs are log-polar images, which are circular by definition so that the last row wraps naturally back to the first row. Thus, circular correlation is desired in this case, so no padding of the images is used before computing the FFTs. In

\[
\text{NCC} = \frac{\mathcal{F}^{-1}(F_1 \cdot F_2^*) - \mathcal{F}^{-1}(F_1 \cdot M_2^\top) \cdot \mathcal{F}^{-1}(M_1 \cdot F_2^*)}{\mathcal{F}^{-1}(F_1 \cdot M_2^\top) \cdot \mathcal{F}^{-1}(M_1 \cdot F_2^*)}
\]

(21)

**Algorithm 2:** Masked registration for translation, rotation, and scale

<table>
<thead>
<tr>
<th>Input:</th>
<th>Fixed image (f_1(x,y)), moving image (f_2(x,y)), fixed mask (m_1(x,y)), moving mask (m_2(x,y))</th>
</tr>
</thead>
<tbody>
<tr>
<td>Output:</td>
<td>Registered moving image (\hat{f}_2(x,y)) and transform parameters (x_0, y_0, \theta_0, \tau)</td>
</tr>
</tbody>
</table>

1. Calculate the Fourier transforms \(F_1(u,v)\) and \(F_2(u,v)\);
2. Calculate the log-polar images of the magnitudes of \(F_1(u,v)\) and \(F_2(u,v)\) to get \(N_1(\log \rho, \theta)\) and \(N_2(\log \rho - \log \tau, \theta - \theta_0)\);
3. Compute the masks of the log-polar images by setting them to 1 inside valid regions and 0 inside invalid regions (see Figure 4);
4. Correlate the log-polar images using masked FFT NCC (without zero padding) to find \(\theta_0\) and \(\tau\);
5. Transform the moving image and moving mask with \(\theta_0\) and \(\tau\) to find \(\hat{f}_2(x,y)\) and \(\tilde{m}_2(x,y)\);
6. Correlate \(f_1(x,y)\) with \(\tilde{f}_2(x,y)\) using masked FFT NCC (with zero padding) with \(m_1(x,y)\) and \(\tilde{m}_2(x,y)\) to find \(x_0\) and \(y_0\);
7. Transform \(\tilde{f}_2(x,y)\) with \(x_0\) and \(y_0\) to find \(\hat{f}_2(x,y)\);

**Fig. 3.** Illustration of the importance of masking even when no mask of the original images is needed. The registration algorithm computes the rotation and scale, transforms the image, and then computes the translation. The scale and rotation transforms can lead to regions on the border where the image information is unknown. Without masking, such regions adversely affect the translation registration. Figure 12 provides several examples on real data.

**Fig. 4.** Illustration of the importance of masking in the polar and log-polar transformation step. In our context, the polar and log-polar transforms are taken on the magnitude of the Fourier transforms, not the original images, but this figure helps illustrate the process. The size of the original image is \((x,y) = [358, 512]\). The second image shows this image in polar coordinates with angle on the y-axis and radius on the x-axis. To avoid invalid regions, the radius is restricted to \(r = \min \left(\frac{x}{2}, \frac{y}{2}\right) = 179\), which yields the largest circle that fits in the image. However, this also ignores many valid regions as shown in the full polar image with radius set to \(r = \sqrt{x^2 + y^2}\). To utilize the valid regions while ignoring the invalid, the mask on the right can be used for masked registration. The same arguments hold for log-polar images.
Fig. 5. Registration using masked FFT NCC versus standard FFT NCC for different translations. Each row represents images with different known transforms which are shown on the left as (x, y, rotation, scale). The standard FFT NCC, influenced by the zero values around the cone beam, often fails to find the correct transform and results in poor correlation scores (shown in white), whereas the masked FFT NCC calculates the transforms correctly with a perfect correlation score of 1.

III. RESULTS

Here we provide several results on real images demonstrating the correctness and effectiveness of our masked FFT NCC algorithm as compared with the standard FFT NCC algorithm. It is not our intention to claim that using the FFT algorithm with the NCC is the best approach, but rather to show the correctness and utility of our embedding of the masking step in the Fourier domain. We demonstrate a range of applications of our method including registration (III-A), image stabilization and object tracking (III-B), generating volumes from serial sections (III-C), and masked registration following rotation and scale transforms (III-D).

A. Ultrasound Masked Registration Results

We used ultrasound images to demonstrate registration results on real images because the masked region is an irregular pie-shape, which demonstrates the ability of the algorithms to work on arbitrary mask shapes. Starting from an original ultrasound image, we transformed the image information by a known amount and then masked both the original and the transformed image with a new window that contains only image information contained in both images.

The results of registering these images with the standard FFT NCC and the masked FFT NCC are given in Figures 5 and 6, which demonstrate translation and rotation/scale results, respectively. The moving images in each row are transformed with different known combinations of translations, rotations, and scaling and then registered to the fixed image. In Figure 5, the masked FFT NCC approach described in Algorithm 1 is applied for translation registration. Notice that the masked results of Figure 5 are able to handle very large transformation misalignments such that the images align in only a small region. In the standard FFT NCC images, the background region has such a large effect on the registration values that it tries to maintain the alignment of the pie-shape, resulting in poor correlation scores as shown in the figures. However, the masked results show that the background region is entirely ignored, and perfect registration results are achieved even in cases where the transform is extreme.

In Figure 6, the enhanced Fourier-Mellin approach of Algorithm 2 is applied to rotation/scale registration. This figure provides examples where the standard Fourier-Mellin approach is insufficient for registering images. As can be seen in the images, the rotation and scale parameters calculated for both the standard and masked versions of the algorithm are the same. However, the invalid regions resulting from the rotation and scaling steps (see Figures 3 and 4) coupled with the original invalid regions themselves greatly influence the subsequent translation registration and lead to failed results in the standard FFT NCC case. On the other hand, the masked translation registration approach accurately handles
these regions to yield correct results. These examples thus demonstrate the advantages of including the masking in the Fourier-Mellin algorithm as described in Section II-D.

For all of the masked results in these two figures, the calculated transform corresponds exactly with the ground truth values and the calculated correlation score is a perfect 1 (except for slight interpolation errors), which is possible since the actual image information is the same in the fixed and moving images except that the moving image pixels are transformed. Note that, in the extreme transform cases, algorithms based on iterative registration would have great difficulty and would likely become stuck in local minima and not converge to the correct solution.

To demonstrate the improved stability of the masked FFT NCC over standard FFT NCC, Figure 7 shows the correlation image corresponding to the image pair in the third row of Figure 5. The white circle shows the true offset location, which is the highest correlation score in the masked FFT NCC image but not in the standard FFT NCC image. This figure also shows that the masked FFT NCC has a smoother decay of the correlation scores from the peak and higher correlation scores overall compared with the standard FFT NCC image.

In contrast, the peak region is more diffuse in the standard correlation image (left) score is the strongest at this location and decays smoothly away from it. In contrast, the peak region is more diffuse in the standard correlation image (right), and it is in the wrong place so that the measured transform is (40, 52) instead of the correct transform of (-100, 100). This error results from the large background regions influencing the correlation score. The correlation values are also much weaker (indicated by color intensity): the maximum score is only 0.62, as opposed to 1.00 in the masked correlation image.

### B. Image Stabilization and Object Tracking

In this section, we demonstrate the use of masked FFT registration for object tracking and image stabilization. We apply our masked FFT NCC algorithm to two distinct translation registration applications of the well-established coast-guard sequence (http://media.xiph.org/video/derf/). A number of frames from the image sequence are shown in Figure 8, which shows two boats passing in the foreground as the camera initially pans to the left (following the small boat) and then to the right (following the large boat). This image sequence has been used in numerous publications including that of Fitch et al. [33] which we described in Section I. In that reference, the authors demonstrated the ability of their algorithm to register the background while ignoring the fast moving boats in the foreground. For example, in Figure 6 of their paper, they generate a mosaic that accurately aligns the background region, and Figure 7 of their paper presents the transform calculated between frames 1 and 80 of the sequence.

Our algorithm enables control over the definition of the mask, and we will demonstrate how different masks can yield very different useful results. We first aim to stabilize the background while the boats move and pass each other in the foreground. A mask can be defined for either the fixed image, the moving image, or both. For a tracking application, we may not be able to define a mask for every frame because we may not know where the regions of interest will be in the following frames. Instead, we can simply define a mask only for the initial frame in the sequence. In our case, we define a mask for the initial frame (fixed image) that is set to 1 in the background and set to 0 in the water region of the image. Since there is a dark vertical line on the right of every image (an artifact), our mask is also set to 0 in that region. The masks for the rest of the frames are set to all ones (no masking).

After defining the mask for the initial frame, we register the first image along with its mask to all of the rest of the images using masked FFT registration. The result is shown in Figure 9, where we have generated a mosaic of the mean of all of the transformed images. In this figure, the background is well registered, and the foreground looks like a blur with the boats almost completely averaged away. We also calculated the transform between frames 1 and 80 and found it to be same values as those calculated by Fitch et al. in [33].

Note that we could have instead registered the images frame-by-frame to attempt more accurate registration since adjacent images have greater overlap. However, this would require the definition of masks at each step since the fixed image would be continually changing.

In this case, we demonstrate the use of masked FFT registration for object tracking and image stabilization. We apply our masked FFT NCC algorithm to two distinct translation registration applications of the well-established coast-guard sequence (http://media.xiph.org/video/derf/). A number of frames from the image sequence are shown in Figure 8, which shows two boats passing in the foreground as the camera initially pans to the left (following the small boat) and then to the right (following the large boat). This image sequence has been used in numerous publications including that of Fitch et al. [33] which we described in Section I. In that reference, the authors demonstrated the ability of their algorithm to register the background while ignoring the fast moving boats in the foreground. For example, in Figure 6 of their paper, they generate a mosaic that accurately aligns the background region, and Figure 7 of their paper presents the transform calculated between frames 1 and 80 of the sequence.

Our algorithm enables control over the definition of the mask, and we will demonstrate how different masks can yield very different useful results. We first aim to stabilize the background while the boats move and pass each other in the foreground. A mask can be defined for either the fixed image, the moving image, or both. For a tracking application, we may not be able to define a mask for every frame because we may not know where the regions of interest will be in the following frames. Instead, we can simply define a mask only for the initial frame in the sequence. In our case, we define a mask for the initial frame (fixed image) that is set to 1 in the background and set to 0 in the water region of the image. Since there is a dark vertical line on the right of every image (an artifact), our mask is also set to 0 in that region. The masks for the rest of the frames are set to all ones (no masking).

After defining the mask for the initial frame, we register the first image along with its mask to all of the rest of the images using masked FFT registration. The result is shown in Figure 9, where we have generated a mosaic of the mean of all of the transformed images. In this figure, the background is well registered, and the foreground looks like a blur with the boats almost completely averaged away. We also calculated the transform between frames 1 and 80 and found it to be same values as those calculated by Fitch et al. in [33].

Note that we could have instead registered the images frame-by-frame to attempt more accurate registration since adjacent images have greater overlap. However, this would require the definition of masks at each step since the fixed image would be continually changing.
Instead of using the algorithm to stabilize the background by setting a mask that ignores the foreground, we can generate an entirely different result by simply changing the initial mask. For example, we can easily track the small boat by defining the mask to be valid only in a region around the small boat. Figure 10 demonstrates the result as the mosaic of the transformed images. Everything around the boat is blurred, but the boat and its wake are crisp and less grainy than the original images.

These examples demonstrate the power and simplicity of the masked FFT registration approach. By simply changing the mask for the initial image in the sequence and then running the algorithm, we can generate entirely different results depending on the desired outcome. Thus, for video applications, we can define different masks for tracking different people, cars, or other objects and then run the algorithm without adjusting any parameters.

C. Generating 3D Volumes from 2D Serial Sections

Another useful application of the masked FFT registration approach is in generating 3D volumes from 2D serial section slices. To generate such images, an entire tissue is thinly sliced and imaged, but movement of the tissue in between slicing steps can lead to misalignment of the resulting images. To correct for this, some form of rigid registration is needed to align the images, assuming that there is sufficient information overlap between adjacent serial sections. In our approach, to build up the 3D volume, each pair of adjacent images is registered using masked FFT registration. Then the transforms are concatenated to find the transformation of every image back to the first image in the sequence. In this way, all images are transformed into the space of the first.

As an example, the top row of Figure 11 shows a sample of the serial sections of an embryo tissue before image alignment. The sagittal and coronal cross-sections clearly show that the images are not well aligned; the anatomical structures in these images have jagged contours and blurred features. Note that these images do not include any regions that need to be masked. However, the invalid regions mapping outside of the image resulting from rotating the images (shown as black regions) do need to be masked (as described in Figures 3 and 4). Without the masking step, all of the black regions would affect the registration metric.

The images in the bottom row of Figure 11 show how clearly the structures can be seen when the 2D serial sections are registered using the translation, rotation, and scale approach of Algorithm 2. Using the masking approach, after each image is rotated, a mask of the invalid regions is stored along with the rotated image, and this mask is used to ignore the black regions in the translation step of the algorithm as illustrated in Figure 3. Figure 12 demonstrates several examples of the difference between using standard FFT registration and masked FFT registration for this step; the invalid regions lead to inaccurate alignment results using the standard FFT registration, and the larger the rotation, the poorer the results. However, the masked FFT registration approach effectively ignores the invalid regions, which demonstrates the power of the masked FFT registration approach, even when the original images do not include regions that need to be masked.

In this particular application, it could be argued that the masked registration could be avoided by setting the mask value to a bright value similar to the background instead of a black value. Unfortunately, this is only viable when the background intensity is a known constant value. However, in real applications such as this one, the background intensity is not known a-priori, and it is also not constant because it depends on the illumination of the image when the picture is taken. Across the set of slices of each volume, the intensity of the background changes, and the variation is even greater from volume to volume. Much care would be required by the experimenter to ensure that the illumination of the background does not vary across the image and is a constant value that could be used for the invalid regions. Fortunately, using masked registration, the general case can be easily solved by ignoring the masked regions entirely.

These results are both significant and exciting because the masking enables a correct processing of the invalid regions.
of the Fourier-Mellin method, a method which is ubiquitous in image analysis applications. In the past, implementations either did not address this concern or used some approach like cropping the image to its largest valid rectangular region, but this has the disadvantage of also eliminating useful valid regions. Instead, using masking, we can entirely ignore the invalid regions and still retain all of the valid regions for accurate registration even in the presence of significant rotations.

D. Masked Registration Following Rotation and Scale Transforms

As described and illustrated in Section III-C, the use of the masked FFT NCC algorithm in the translation registration stage of the Fourier-Mellin algorithm overcomes a number of important limitations of that algorithm. When incorporating masking into this algorithm, after each image is rotated and scaled, a mask of the invalid regions is stored along with the rotated image, and this mask is used to ignore the invalid regions in the translation step of the algorithm as illustrated in Figure 3. In this section, we demonstrate additional quantitative and qualitative examples.

For these experiments, we utilized the images from the INRIA dataset that were used in the paper and supplementary material of Tzimiropoulos et al. in [23]. These datasets were obtained from [39]. This dataset contains a range of images of natural scenes where the camera is significantly rotated, scaled, and translated. Four example scenes are shown in Figure 13. Each of these scenes was imaged under a variety of transforms as described in the results of [23] and its supplementary material.

To test the impact of masking for the translation registration after rotation and scale transforms, we transform the images using the given transform and compare the results of computing the translation registration using standard FFT NCC and masked FFT NCC. When the images are rotated and scaled, this can lead to significant invalid regions surrounding the image, as illustrated by the red regions around the transformed images in the last column of Figure 13. These invalid regions adversely affect the standard FFT NCC computation and lead to a large number of failures, some of which can be seen in the third column of that figure. In contrast, the masked FFT NCC ignores these regions and accurately computes the translation component that leads to the optimal value of the NCC metric as shown in the last column. The overlaid images for the masked FFT NCC results are nearly indistinguishable from those in the supplementary material of [23].

Table II presents quantitative results comparing the calculated translation component to the ground truth transform provided with the images in [39]. The table demonstrates that the masked FFT NCC transforms are very close to the ground truth for all of the images, whereas the standard FFT NCC fails in many cases. The masked FFT NCC also achieves much higher maximum NCC values than the corresponding standard FFT NCC (more than double in most cases) because of its ability to ignore the masked-out regions.

E. Timing Results

The algorithms were implemented both in ITK and Matlab. Some implementations of the FFT computation require that the images be extended with zeros to a common power of two. The VNL library used in ITK is less restrictive in that it allows matrix sizes that are multiples of 2, 3, and 5. Other libraries such as FFTW (www.fftw.org) that is used in Matlab can operate on sequences of any length, but such FFTs favor sizes with small prime factors such as multiples of 2, 3, and 5. Therefore, once we have calculated the correlation matrix size (one less than the sum of the sizes of the images), we include a routine for calculating the next largest size that consists of multiples of these numbers. We accomplish this by simply adding one until the size can be factored by those numbers.
We then extend the fixed and moving images to this size by padding with zeros before computing the FFTs.

The timing tests were run on a 3 GHz Dell computer with 2 GB of RAM. Figure 14 illustrates the masked FFT NCC time for a range of image sizes. For example, the algorithm takes about 0.85 seconds for a typical medical image of size $512 \times 512$ or 3.4 seconds for a typical microscope image of size $1024 \times 1024$. For comparison, we also show the timing plot from the Matlab built-in function normxcorr2, which uses integral images as described in [15]. The time required for the masked FFT NCC is only slightly greater than that of normxcorr2 even though normxcorr2 cannot handle masking and uses a simplified version of NCC whose assumptions are violated for images of the same size so that the resulting correlation is incorrect (see www.mathworks.com/matlabcentral/fileexchange/29005-generalized-normalized-cross-correlation for more details and a solution to this problem). This demonstrates the computational efficiency of our approach.

We have also been collaborating with researchers who have extended our algorithm for sub-pixel registration and implemented our algorithm on a GPU using CUDA. They have reported that the resulting program can register hundreds of frames per second for $512 \times 512$ images on recent GPUs for the application of video registration. We plan to report further details as these results are refined.

Fig. 13. Results on four scenes taken from [23] and [39]. In those references, these scenes were named (from top to bottom): Crolle, Bip, Boat, and Resid. The first two columns show pairs of significantly transformed images, where the scale and rotation pairs $(s, \theta)$ are, from top to bottom, $(3.97, 0^\circ)$, $(2.69, 0^\circ)$, $(4.3, 46^\circ)$, and $(5.9, -32^\circ)$. Given these transforms, the standard FFT NCC and masked FFT NCC were applied to correct the translational components between the images. The large invalid regions resulting from the rotation and scale often cause the standard FFT NCC to fail, whereas our masked algorithm computes the transforms correctly. The masked results correspond closely to the ground truth transforms reported in [23]. Table II gives the quantitative results for all pairs of transforms for these scenes.

<table>
<thead>
<tr>
<th>Scene</th>
<th>Masked Registration Time (in seconds)</th>
<th>Standard FFT NCC</th>
<th>Matlab normxcorr2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Crolle</td>
<td>0.50</td>
<td>2.20</td>
<td>3.30</td>
</tr>
<tr>
<td>Bip</td>
<td>0.65</td>
<td>2.91</td>
<td>4.02</td>
</tr>
<tr>
<td>Boat</td>
<td>0.70</td>
<td>3.23</td>
<td>4.34</td>
</tr>
<tr>
<td>Resid</td>
<td>0.75</td>
<td>3.54</td>
<td>4.65</td>
</tr>
</tbody>
</table>

Fig. 14. Masked registration time for different image sizes. The blue plot shows the masked FFT NCC time, and, for comparison, the red plot shows the time for the Matlab built-in function normxcorr2. Although the normxcorr2 function is slightly faster, it does not enable masking. The x-axis represents the side length of a 2D image, and the y-axis provides the registration time in seconds.

IV. CONCLUSIONS AND FUTURE WORK

The main contribution of this paper is the derivation of a mathematical framework that enables the efficient translation registration in the Fourier domain of images that have associated masks. By defining the masked registration in the Fourier domain, the algorithm is able to take advantage of the fast, global, and parameter-free advantages of Fourier domain registration. We also demonstrated that the computation of the masked registration is as efficient as the standard FFT NCC registration, requiring only 3 forward FFTs, 6 backward FFTs, and a number of element-wise matrix multiplications.
The results demonstrate the correctness of the algorithm as well as its applicability to a large range of applications.

The fundamental framework of this approach suggests many possible extensions for future work. The algorithms are not inherently limited to intensity images, and it should be straightforward to extend them to gradient images such as those used in the orientation correlation [22] and normalized gradient correlation [23] approaches. We are also currently looking into extensions to enable sub-pixel accuracy. Other topics include extending the approach to multi-modality, similar to the approach taken in [34], or to different correlation methods such as phase correlation. Finally, an extension to masked FFT deformable registration would enable a whole new range of applications.

REFERENCES


Dirk Padfield earned his BS degrees in electrical engineering and international studies in 2000 and his MS degree in electrical engineering in 2002 from the Pennsylvania State University. He completed his Ph.D. in Computer Science in 2009 at the Rensselaer Polytechnic Institute. Since 2002 he has worked as a Lead Scientist at the GE Global Research Center in Niskayuna, NY where he has developed novel segmentation, registration, and tracking algorithms for a broad range of applications in biological, medical, and industrial imaging modalities including microscopy, ultrasound, PET, MR, and CT. He has published over 25 technical peer-reviewed journal and conference publications, has authored more than 10 patents, and has transitioned numerous algorithms into medical and industrial products.