Applications of 1-D Versions of Image Referencing Techniques to Hydrology and to Patient Rehabilitation

Roberto Araiza, Martine Ceberio,
Naga Suman Kanagala,
and Vladik Kreinovich
Department of Computer Science
University of Texas at El Paso
El Paso, TX 79968
Contact email vladik@utep.edu

Gang Xiang
Philips Healthcare Informatics
El Paso, TX 79912
Email: gxiang@acm.org

***Abstract*—In this paper, we consider two seemingly unrelated problems: the hydrology problem of relation between groundwater and surface water, and a problem of identification of human gait in neuro-rehabilitation. It turns out that in both problems, we can efficiently use soft computing-motivated algorithms originally developed for image referencing.**

I. First Practical Problem: Relation Between Groundwater and Surface Water

Most of our water consumption comes from groundwater reservoirs. Because of this, it is vitally important, for each project – whether it is a new water-consuming manufacturing plant or a new way of agriculture irrigation – to predict how this project will affect the groundwater. One way groundwater is affected is through the surface water: contamination of surface water can lead to an eventual contamination of groundwater. The relation between groundwater and surface water is often very complex. In some geographic areas, the groundwater reservoirs are pretty secure; whatever surface water seeps into these reservoirs, it takes a long time, during which many contaminants are safely dissolved. In other areas, the relation is more straightforward, and the contaminants in the surface water take a very short time to propagate into the groundwater reservoirs.

In view of this possible difference, it is desirable to determine a delay t_0 between the change in the surface water and the resulting change in the groundwater. One way to find this time delay is to measure the characteristics $x(t)$ of the surface water at different moments of time, and to measure the characteristics $x'(t)$ of the ground water. Because of the delay, the characteristics of the groundwater $x'(t)$ at a time t are closely related with the state of the surface water at time $t - t_0$:

$$x'(t) \approx x(t - t_0).$$

The question is how to find the delay t_0 once we know the time series $x(t)$ and $x'(t)$; see, e.g., [5].

One possibility is to try all possible shifts t_0 but this is computationally very intensive. In this talk, we adjust the known soft-computing-motivated fast image referencing techniques (see, e.g., [4]) to design a fast algorithm that uses Fast Fourier Transform for finding the optimal shift.

II. Second Practical Problem: Identification of Human Gait in Neuro-Rehabilitation

Many neurological diseases such as stroke, traumatic body injury, and spinal cord injury drastically decrease the patient's ability to walk without physical assistance. To re-establish normal gait, patients undergo extensive rehabilitation. At present, rehabilitation requires gait assessment by highly qualified experienced clinicians. To make rehabilitations easier to access and to decrease the rehabilitation cost, it is desirable to automate gait assessment; see, e.g., [3].

In precise terms, a gait is measured by the dependence $x'(t)$ of some characteristic – e.g., the acceleration or the angle between different parts of the foot. The gait assessment means comparing the recorded patient's gait with a standard (average) gait $x(t)$ of healthy people of the same age, body measurements, etc. One of the problems in this comparison is that patients walk slower; so, to properly compare gaits, we must first appropriately shift and "scale" the standard gait so that it best matches the speed with which the patient walks, i.e., to find the values t_0 and λ for which

$$x'(t) \approx x(\lambda \cdot t - t_0).$$

One possibility is to try all possible shifts and scalings but this is computationally very intensive. In this talk, we adjust the known image referencing techniques to design an efficient algorithm that uses Fast Fourier Transform for finding the optimal combination of a shift and a scaling.

III. Image Referencing (Registration) Problem: A Brief Reminder

In geosciences, we often need to combine two (or more) images of the same area:

- different images bring different information; so, to get a better understanding, we must *fuse* the corresponding data; e.g., we must combine a satellite image with a radar image;
- comparison of two images – e.g., images made at different moments of time – can also give us information about the changes: e.g., by comparing pre- and post-earthquake images, we can determine the effect of the earthquake.

Compared images are often obtained from slightly different angles, from slightly different positions. Therefore, in order to compare these images, we must first *reference* (*register*) them, i.e., find the shift, rotation, and scaling after which these images match as much as possible, and then apply these transformations to the original images.

IV. Image Registration Algorithms: Brief Overview

There exist many methods for image registration; see, e.g., [1], [6]. Among the most widely used methods are methods of *point matching*, where we find the matching points in the two images, and then the most appropriate transformation (rotation and/or shift) which maps the points from one image into the corresponding points from the other image.

Point matching methods work well when the images have clearly identifiable matching points, and when we know the images with a high accuracy – so that we can identify and match these matching points with a reasonable accuracy. For example, in satellite images, we often have clear matching points representing special landmarks such as landmark city areas, landmark bridges, or tips of peninsulas. Such landmarks can usually be easily found in highly populated areas or in special terrains in which the area is highly non-homogeneous: e.g., there may be a clear shore line with a clear landmark point, or there may be a large clearly distinguishable river with a clear landmark turn.

However, there are many homogenous areas where it is not easy to find landmarks. For example, in the desert areas of the U.S. Southwest, the only visible landmarks are road intersections, and there are usually several similar-looking road intersections in the same image, so it is difficult to find the matching points between the two images. Similarly, in the mountain areas, there are many landmarks like summits and ridges, but usually, there are several similar-looking summits and ridges in each image, so it is difficult to match points in the two images.

For images known with very low accuracy, we may still find landmarks. However, since we only know the images with a very low accuracy, we may only be able to locate these landmarks with a very low accuracy, too low to enable us to adequately register the two images.

Sometimes, instead of landmark *points*, we have landmark *features*. For example, we may not have a landmark bridge, but we may have a clearly distinguishable river. In such situations, instead of matching points, we can match features. Such feature-matching algorithms are also efficiently used in image registration. However, in homogenous terrains and/or in situations when we only know the images with low accuracy, we may only be able to locate these features with a very low accuracy, too low to enable us to adequately register the two images.

In some cases, e.g., in many astronomical images, we have an image surrounded by an empty space. In this case, even when we cannot find the landmark points in the two images, we can match these images by comparing, e.g., the centers of gravity of these images. Alas, this is not the case in images like satellite images or radar images.

As an example of low accuracy images for which registration is practically important, we will actually consider low accuracy satellite images. So, in order to come up with an algorithm for registering low accuracy images, an algorithm which should be applicable for satellite images, we must place our emphasis on image registration techniques which go beyond point matching, feature matching, or simple geometric transformations in the image domain. Many such algorithms are based on the use of the Fast Fourier Transform (FFT).

Before we start describing these methods and adjusting them to the signal, let us provide a motivation for using FFT in such problems.

V. Why Fourier-Based Methods

Let us start with a hydrology-motivated problem. We have two functions $x(t)$ and $x'(t)$, and we must find the shift t_0 for which the signal $x'(t)$ will be, in some reasonable sense, the closest to the shifted signal $x(t-t_0)$. A reasonable way to describe the closeness between the two signals is to require that for every moment of time t, the corresponding intensities are close to each other. We can use, e.g., the squared difference $(x'(t)-x(t-t_0))^2$ between these values as the measure of the similarity at t, and we can use the sum (integral) $\int (x'(t)-x(t-t_0))^2\,\mathrm{d}t$ of these square differences over all moments t as the measure of overall similarity between the two signals.

The problem of finding the shift t_0 that minimizes the above integral takes the following form: find t_0 for which the integral $\int (x'(t)-x(t-t_0))^2\,\mathrm{d}t$ attains the smallest possible value. By representing the square of the difference $(x'-x)^2$ as the sum of three terms $(x')^2+x^2-2\cdot x'\cdot x$, we can represent the above scoring function as

$$\int (x'(t))^2\,\mathrm{d}t + \int x(t-t_0)^2\,\mathrm{d}t - 2\int x'(t)\cdot x(t-t_0)\,\mathrm{d}t.$$

The first integral in the sum does not depend on the shift at all. By using the new variable $s=t-t_0$, we can show that the second integral is equal to $\int x(s)^2\,\mathrm{d}s$ and thus, also does not depend on the shift. So, finding the shift for which the sum is the smallest possible is equivalent to finding the shift for which the *cross-correlation* term $\int x'(t)\cdot x(t-t_0)\,\mathrm{d}t$ attains the largest possible value.

For signals described by values at n moments of time, a straightforward approach would require that we compute the value of the scoring function for all n possible shifts t_0. Computing each integral requires time $O(n)$, so overall, we need time $O(n)\cdot O(n)=O(n^2)$.

This computation can be performed much faster if we take into account that the cross-correlation term is a *convolution* between the signals $x(t)$ and $x'(t)$. Convolution is one of the main techniques in signal processing, and it is well known that we can compute convolution faster (in time $O(n \cdot \log(n)))$ by using Fast Fourier Transform (FFT); see, e.g., [2]. Specifically, to compute the convolution, we need the following steps:

- first, we apply FFT to the original signals, resulting in functions $F(\omega)$ and $F'(\omega)$;
- then, for each frequency ω, we compute the product
$$R(\omega) \stackrel{\text{def}}{=} F(\omega) \cdot (F')^*(\omega)$$
(where F^* means complex conjugation);
- finally, we apply the inverse Fourier transform to the resulting function $R(\omega)$, and get the desired auto-correlation function.

We can now find the shift as the value t_0 for which the cross-correlation attains the largest possible value.

The FFT of a signal of size n requires $O(n \cdot \log(n))$ steps. Multiplication of the two Fourier transforms and the final search for the largest value both require processing each value ω and t_0 once, so both require time $O(n)$. As a result, we can find the desired shift t_0 in time $O(n \cdot \log(n)) + O(n) = O(n \cdot \log(n))$.

VI. The Simplest Case: Shift Detection in the Absence of Noise

Let us first consider the above case when two signals differ only by shift. It is known that if two signals $x(t)$ and $x'(t)$ differ only by shift, i.e., if $x'(t) = x(t - t_0)$ for some (unknown) shift t_0, then their Fourier transforms

$$F(\omega) = \frac{1}{\sqrt{2\pi}} \cdot \int x(t) \cdot e^{-2\pi \cdot \mathrm{i} \cdot (t \cdot \omega)} \, \mathrm{d}t,$$

$$F'(\omega) = \frac{1}{\sqrt{2\pi}} \cdot \int x'(t) \cdot e^{-2\pi \cdot \mathrm{i} \cdot (t \cdot \omega)} \, \mathrm{d}t,$$

where $\mathrm{i} \stackrel{\text{def}}{=} \sqrt{-1}$, are related by the following formula:

$$F'(\omega) = e^{2\pi \cdot \mathrm{i} \cdot (-\omega \cdot t_0)} \cdot F(\omega). \tag{1}$$

Let us explain this formula. Since $x'(t) = x(t - t_0)$, the Fourier transform $F'(\omega)$ of the image $x'(t)$ takes the form

$$F'(\omega) = \frac{1}{\sqrt{2\pi}} \cdot \int x(t - t_0) \cdot e^{-2\pi \cdot \mathrm{i} \cdot (t \cdot \omega)} \, \mathrm{d}t.$$

We can simplify this expression if we introduce a new variable $s \stackrel{\text{def}}{=} t - t_0$, so that $t = s + t_0$. Here, $\mathrm{d}s = \mathrm{d}t$, so

$$F'(\omega) = \frac{1}{\sqrt{2\pi}} \cdot \int x(s) \cdot e^{-2\pi \cdot \mathrm{i} \cdot ((s + t_0) \cdot \omega)} \, \mathrm{d}s.$$

Here, $((s + t_0) \cdot \omega) = (s \cdot \omega) + (t_0 \cdot \omega)$, hence

$$e^{-2\pi \cdot \mathrm{i} \cdot ((s + t_0) \cdot \omega)} = e^{-2\pi \cdot \mathrm{i} \cdot (s \cdot \omega)} \cdot e^{-2\pi \cdot \mathrm{i} \cdot (t_0 \cdot \omega)}.$$

The second factor does not depend on s, so we can move it outside the integral and conclude that

$$F'(\omega) = e^{-2\pi \cdot \mathrm{i} \cdot (t_0 \cdot \omega)} \cdot \left(\frac{1}{\sqrt{2\pi}} \cdot \int x(s) \cdot e^{-2\pi \cdot \mathrm{i} \cdot (s \cdot \omega)} \, \mathrm{d}s \right).$$

The expression in parentheses is exactly $F(\omega)$, so indeed the formula (1) is true.

It is known that the magnitude $|e^{-2\pi \cdot \mathrm{i} \cdot (t_0 \cdot \omega)}|$ (also known as the *modulus* or the *absolute value*) of the complex value

$$e^{-2\pi \cdot \mathrm{i} \cdot (\omega \cdot t_0)} = \cos(-2\pi \cdot (\omega \cdot t_0)) + \mathrm{i} \cdot \sin(-2\pi \cdot (\omega \cdot t_0))$$

is equal to 1. Therefore, if the images are indeed obtained from each other by shift, then their Fourier transforms have the same magnitude:

$$M'(\omega) = M(\omega), \tag{2}$$

where we denoted

$$M(\omega) = |F(\omega)|, \quad M'(\omega) = |F'(\omega)|. \tag{3}$$

The actual value of the shift t_0 can be obtained if we use the formula (1) to compute the value of the following ratio:

$$R_0(\omega) = \frac{F'(\omega)}{F(\omega)}. \tag{4}$$

Substituting (1) into (4), we get

$$R_0(\omega) = e^{-2\pi \cdot \mathrm{i} \cdot (\omega \cdot t_0)}. \tag{5}$$

Therefore, the inverse Fourier transform $P_0(t)$ of this ratio is equal to the delta-function $\delta(t + t_0)$.

In other words, in the ideal no-noise situation, this inverse Fourier transform $P_0(t)$ is equal to 0 everywhere except for the point $t = -t_0$; so, from $P_0(t)$, we can easily determine the desired shift by using the following *algorithm*:

1) we apply FFT to the original signals $x(t)$ and $x'(t)$ and compute their Fourier transforms $F(\omega)$ and $F'(\omega)$;
2) we compute the ratio $R_0(\omega)$;
3) we apply the inverse FFT to the ratio $R_0(\omega)$ and compute its inverse Fourier transform $P_0(t)$;
4) we determine the desired shift t_0 as the only value t_0 for which $P_0(-t_0) \neq 0$.

VII. Shift Detection in the Presence of Noise

The above simplified algorithm assumes that the signals $x(t)$ and $x'(t)$ are exactly the same signal, differing only by a shift: $x'(t) = x(t - t_0)$. In real life, e.g., the measured intensity of the groundwater values do not depend only on the surface water levels, so the corresponding intensity values will be only approximately equal: $x'(t) \approx x(t - t_0)$.

In the ideal non-noise case, the inverse Fourier transform $P_0(t)$ of the ratio (4) is equal to the delta-function

$$\delta(t + t_0),$$

i.e., equal to 0 everywhere except for the point $t = -t_0$. In the presence of noise, the values of $P_0(t)$ will be slightly different from the delta-function. It seems reasonable to expect that still, the value $|P_0(-t_0)|$ should be much larger than all the other values of this function. Thus, in principle, it may seem that the value of the shift can be determined as the value at which $|P_0(-t_0)|$ is the largest.

In practice, however, due to noise, for some frequencies ω, the value of the Fourier transform $F(\omega)$ corresponding to the signal $x(t)$ may be close to 0, while the value of the Fourier transform $F'(\omega)$ corresponding to the signal $x'(t)$ may be non-zero. For such frequencies, the ratio (4) can be very high. These high values dominate the ratio $R_0(\omega)$ and thus, distort the inverse Fourier transform $P_0(t)$. To avoid this distortion, it is desirable to replace the formula (4) with a more noise-resistant one.

In general, one of the general techniques for making a data processing algorithm more noise-resistant is to take into account constraints on the input data. In the ideal case, the magnitude $|R_0(\omega)|$ of the complex ratio $R_0(\omega)$ (as described by the expression (4)) is equal to 1. In the presence of noise, the observed values of the intensities may differ from the actual values; as a result, their Fourier transforms also differ from the values and hence, the magnitude of the ratio (4) may be different from 1.

Let us therefore describe how we can improve the accuracy of this method if, instead of simply processing the measurement results, we take into consideration the additional knowledge that the magnitude of the actual ratio (4) is exactly equal to 1.

Let us denote the actual (unknown) value of the value $e^{-2\pi\cdot\mathrm{i}\cdot(\omega\cdot t_0)}$ by r. Then, in the absence of noise, the equation (1) takes the form

$$F'(\omega) = r \cdot F(\omega). \tag{5}$$

In the presence of noise, the computed values $F(\omega)$ and $F'(\omega)$ of the Fourier transforms can be slightly different from the actual values, and therefore, the equality (5) is only approximately true:

$$F'(\omega) \approx r \cdot F(\omega). \tag{6}$$

In addition to the equation (6), we know that the magnitude of r is equal to 1, i.e., that

$$|r|^2 = r \cdot r^* = 1, \tag{7}$$

where r^* denotes a complex conjugate to r.

As a result, we know two things about the unknown value r:

- that r satisfies the approximate equation (6), and
- that r satisfies the additional constraint (7).

We would like to get the best estimate for r among all estimates that satisfy the condition (7). To get the optimal estimate, we can use the *Least Squares Method* (LSM). According to this method, for each estimate r, we define the error

$$E = F'(\omega) - r \cdot F(\omega) \tag{8}$$

with which the condition (6) is satisfied. Then, we find among all estimates which satisfy the additional condition (7), a value r for which the square $|E|^2 = E \cdot E^*$ of this error is the smallest possible.

The square $|E|^2$ of the error E can be reformulated as follows:

$$E \cdot E^* = (F'(\omega) - r \cdot F(\omega)) \cdot \left(F'^*(\omega) - r^* \cdot F^*(\omega)\right) =$$
$$F'(\omega) \cdot F'^*(\omega) - r^* \cdot F^*(\omega) \cdot F'(\omega) - r \cdot F(\omega) \cdot F'^*(\omega) +$$
$$r \cdot r^* \cdot F(\omega) \cdot F^*(\omega). \tag{9}$$

We need to minimize this expression under condition (7).

For conditional minimization, there is a known technique of Lagrange multipliers, according to which the minimum of a function $f(x)$ under the condition $g(x) = 0$ is attained when for some real number λ, the auxiliary function $f(x) + \lambda \cdot g(x)$ attains its unconditional minimum; this value λ is called a *Lagrange multiplier*.

For our problem, the Lagrange multiplier technique leads to the following unconditional minimization problem:

$$\text{Minimize} F'(\omega) \cdot F'^*(\omega) - r^* \cdot F^*(\omega) \cdot F'(\omega) -$$
$$r \cdot F(\omega) \cdot F'^*(\omega) + r \cdot r^* \cdot F(\omega) \cdot F^*(\omega) +$$
$$\lambda \cdot (r \cdot r^* - 1). \tag{10}$$

We want to find the value of the complex variable r for which this expression takes the smallest possible value. A complex variable is, in effect, a pair of two real variables, so the minimum can be found as a point at which the partial derivatives with respect to each of these variables are both equal to 0. Alternatively, we can represent this equality by computing the partial derivative of the expression (10) relative to r and r^*. If we differentiate (10) relative to r^*, we get the following linear equation:

$$-F^*(\omega) \cdot F'(\omega) + r \cdot F(\omega) \cdot F^*(\omega) + \lambda \cdot r = 0. \tag{11}$$

From this equation, we conclude that

$$r = \frac{F^*(\omega) \cdot F'(\omega)}{F(\omega) \cdot F^*(\omega) + \lambda}. \tag{12}$$

The coefficient λ can now be determined from the condition that the resulting value r should satisfy the equation (7). In other words, we must have

$$\frac{|F^*(\omega) \cdot F'(\omega)|}{|F(\omega) \cdot F^*(\omega) + \lambda|} = 1,$$

i.e., equivalently, that

$$|F^*(\omega) \cdot F'(\omega)| = |F(\omega) \cdot F^*(\omega) + \lambda|. \tag{13}$$

The expression $F(\omega) \cdot F^*(\omega) + \lambda$ is a real number, so – depending on the sign – its magnitude (absolute value) is equal either to this same number or to its opposite, i.e.,

$$F(\omega) \cdot F^*(\omega) + \lambda = \pm|F(\omega) \cdot F^*(\omega) + \lambda|. \tag{14}$$

Due to (13) and (14), we thus have

$$F(\omega) \cdot F^*(\omega) + \lambda = \pm |F^*(\omega)| \cdot |F'(\omega)|. \tag{15}$$

Substituting the expression (15) into the formula (11), we conclude that

$$r = \pm \frac{F^*(\omega) \cdot F'(\omega)}{|F^*(\omega)| \cdot |F'(\omega)|}.$$

In principle, the sign can depend on the frequency ω. However, since the observed signals $x(t)$ and $x'(t)$ are functions which are different from 0 only in a bounded area, their Fourier transforms are continuous. It is therefore reasonable to consider expressions which are continuously depending on the frequency ω. To make the above expression continuous, we must use the same sign for all frequencies. If we use the positive sign for all the frequencies, then we arrive at the following ratio:

$$r = \frac{F^*(\omega) \cdot F'(\vec{\omega})}{|F^*(\vec{\omega})| \cdot |F'(\vec{\omega})|}. \tag{16}$$

(One can check that if we select a negative sign for all the frequencies, we will end up with the exact same algorithm.)

So, in the presence of noise, instead of using the ratio (4), we should compute, for every ω, the optimal approximation

$$R(\omega) = \frac{F^*(\omega) \cdot F'(\omega)}{|F^*(\omega)| \cdot |F'(\omega)|}. \tag{17}$$

This expression is known in signal and image processing; it is called a "cross-correlation power spectrum" (see, e.g., [1], [6]). What we have just shown is that this expression provides an optimal estimate for the desired value $e^{-2\pi \cdot \mathrm{i} \cdot (\omega \cdot t_0)}$.

How can we use this ratio in registration? In the ideal non-noise case, when $x'(t) = x(t - t_0)$, the ratio (17) also equals to $e^{-2\pi \cdot \mathrm{i} \cdot (\omega \cdot t_0)}$. Hence, in the ideal no-noise case, the inverse Fourier transform $P(t)$ of this ratio is equal to the delta-function $\delta(t + t_0)$, i.e., it is equal to 0 everywhere except for the point $t = -t_0$. In the presence of noise, we expect the values of $P(t)$ to be slightly different from the delta-function, but still, the value $|P(-t_0)|$ should be much larger than all the other values of this function. Thus, the value of the shift can be determined as the value at which $|P(-t_0)|$ is the largest. We arrive at the following algorithm:

1) we apply FFT to the original signals $x(t)$ and $x'(t)$ and compute their Fourier transforms $F(\omega)$ and $F'(\omega)$;
2) we compute the ratio $R_0(\omega)$;
3) we apply the inverse FFT to the ratio $R_0(\omega)$ and compute its inverse Fourier transform $P_0(t)$;
4) we determine the desired shift t_0 as the only value t_0 for which the magnitude $|P(-t_0)|$ attains the largest possible value.

VIII. The Role of Soft Computing: A Comment

In the above explanation of why FFT-based techniques are useful in shift detection, we started with the need to describe the degree with which the signals $x'(t)$ and $x(t - t_0)$ are similar, and ended up with an integral expression

$$\int (x'(t) - x(t - t_0))^2 \, \mathrm{d}t.$$

To come up with this expression, we simply made a heuristic transition. A more justified way would be to use a formalism which is specifically designed to translate natural-language expressions like "close" and "similar" to explicit expressions – the formalism of *fuzzy logic*.

For example, to describe the function "similar", we can use a Gaussian membership function

$$\mu(x' - x) = \exp(-c \cdot (x' - x)^2),$$

for some real number $c > 0$. In this case, for each moment of time t, the degree with which the corresponding values $x'(t)$ and $x(t - t_0)$ are similar is equal to

$$\exp(-c \cdot (x'(t) - x(t - t_0))^2).$$

We want to find the degree to which the corresponding values are similar for the first moment of time $t^{(1)}$, *and* the corresponding values are similar for the second pixel $t^{(2)}$, etc.

If we use the algebraic product $a \cdot b$ to describe "and" – one of the choices proposed in Zadeh's original papers – then the resulting degree that the images $x'(t)$ and $x(t - t_0)$ are similar is equal to the product

$$\prod_t \exp(-c \cdot (x'(t) - x(t - t_0))^2).$$

Since the product of the exponents is equal to the exponent of the sum, this degree is equal to $\exp(-c \cdot S)$, where

$$S \stackrel{\text{def}}{=} \sum_t (x'(t) - x(t - t_0))^2.$$

Since the degree of similarity $\exp(-c \cdot S)$ is a monotonically decreasing function of S, to find the shift t_0 for which the degree of similarity between the signal $x'(t)$ and the shifted signal $x(t - t_0)$ is the largest, we must find the shift for which the sum S is the smallest. This sum is, in effect, the desired integral $\int (x'(t) - x(t - t_0))^2 \, \mathrm{d}t$.

We have selected this quadratic expression simply to explain that, even in this simple setting, we need to use FFT to make shift determination more efficient. If this quadratic integral was the most adequate expression of the degree of similarity between the two images, then we would be able to argue that, since this expression is an example of least square expressions used in statistical data processing, we could probably use the traditional statistical techniques to derive this expression.

However, it is known that in many practical problems, the above more sophisticated FFT-based algorithm works much better than the minimization of the quadratic integral (see, e.g., [1]) – which shows that the quadratic integral is, in general, not necessarily the most adequate description of image matching.

Intuitively, it is reasonably clear why the more sophisticated algorithm is often better: in the ideal case of the exact shift, this algorithm returns an impulse function, which has 0 values everywhere except for the desired shift point $-t_0$, and which has a high value at $-t_0$. In contrast, the value of the cross-correlation function is slowly decreasing around $t = -t_0$. Not surprisingly, in the presence of a reasonable noise, the new algorithm leads to a more accurate reconstruction of the shift than the method based on the minimization of the quadratic integral.

The fact that a more sophisticated expression for the degree of similarity is needed is a good indication that soft computing techniques are needed here: for soft computing, if we select more complex membership functions and more complex "and" operations (t-norms), we would end up with more complex expression for the degree of similarity between the two images.

IX. Reducing Scaling to Shift

Let us now consider a more complex problem, in which we must find a shift t_0 and scaling λ for which

$$x'(t) \approx x(\lambda \cdot t - t_0).$$

Since, in addition to shift, we also have scaling, the magnitudes $M(\omega)$ and $M'(\omega)$ of the corresponding Fourier transforms are not equal, but differ from each by the corresponding scaling:

$$M'(\omega) \approx \frac{1}{\lambda} \cdot M\left(\frac{\omega}{\lambda}\right).$$

If we go to *log frequencies* $\rho = \log(\omega)$ (for which $\omega = \exp(\rho)$), then scaling becomes shift-like: $\rho \to \rho - b$, where $b = \log(\lambda)$. So, in log frequencies, scaling is described by a shift.

In view of the above reduction, in order to determine the scaling between M and M', we can do the following:

- transform both images from the original frequencies to log frequencies;
- use the above FFT-based algorithm to determine the corresponding shift $\log(\lambda)$;
- from the corresponding "shift" values, reconstruct the scaling coefficient λ.

Comment. The main computational problem with the transformation to log frequencies is that we need values $M(\xi)$ on a rectangular grid in log frequencies space, but computing $\log(\omega)$ for the original grid points leads to points outside that grid. So, we need interpolation to find the values $M(\xi)$ on the desired grid. One possibility is to use *linear* interpolation.

X. Final Algorithm: Determining Shift and Scaling

1) we apply FFT to the original signals $x(t)$ and $x'(t)$ and compute their Fourier transforms $F(\omega)$ and $F'(\omega)$;
2) we compute the magnitudes $M(\omega) = |F(\omega)|$ and $M'(\omega) = |F'(\omega)|$ of these Fourier transforms;
3) we apply the above scaling detection algorithm to the functions $M(\omega)$ and $M'(\omega)$, and determine the scaling coefficient λ;
4) we apply the corresponding scaling to $x(t)$; as a result, we get a new signal $\widetilde{x}(t)$;
5) the signals $\widetilde{x}(t)$ and $x'(t)$ are already aligned in terms of scaling, the only difference between them is in an (unknown) shift; so, we again apply the above described FFT-based algorithm for determining shift: this time, actually to determine shift.

As a result, we get the desired values of shift and scaling. This algorithm also requires the time $O(n \cdot \log(n))$. hence, we get the desired registration.

Acknowledgments.

This work was supported in part by NSF grants HRD-0734825, EAR-0225670, and EIA-0080940, by Texas Department of Transportation grant No. 0-5453, by the Japan Advanced Institute of Science and Technology (JAIST) International Joint Research Grant 2006-08, and by the Max Planck Institut für Mathematik.

The authors are thankful to the anonymous referees for valuable discussions.

References

[1] L. G. Brown, "A Survey of Image Registration Techniques", *ACM Computing Surveys*, 1992, Vol. 24(4), pp. 325–376.

[2] Th. Cormen, C. E. Leiserson, R. V. Rivest, and C. Stein, *Introduction to Algorithms*, MIT Press, Cambridge, Massachusetts, 2001.

[3] T. Sarkodie-Gyan, *Neuro-rehabilitation devices: Engineering Design, Measurement, and Control*, McGraw-Hill, 2005.

[4] C. G. Schiek, R. Araiza, J. M. Hurtado, A. A. Velasco, V. Kreinovich, and V. Sinyansky, "Images with Uncertainty: Efficient Algorithms for Shift, Rotation, Scaling, and Registration, and Their Applications to Geosciences", In: M. Nachtegael, D. Van der Weken, E. E. Kerre, and W. Philips (eds.), *Soft Computing in Image Processing: Recent Advances*, Springer Verlag, 2007, pp. 35–64.

[5] Z. Sheng, A. McDonald, G. Xiang, and C. Hart, "Characterize interaction of surface and ground water along the Pecos River by using time-series analysis of observed water levels", *Proceedings of the International Conference on Hydrological Sciences for Managing Water Resources in Asian Developing World*, Guangzhou, China, June 8–10, 2006.

[6] B. Zitová and J. Flusser, "Image registration methods: a survey", *Image and Vision Computing*, 2003, Vol. 21, pp. 977–1000.