Biological Pattern Matching

A Novel Pattern Matching Technique based on Fourier Transforms

Saurabh V. Pendse

The Maharaja Sayajirao University of Vadodara

March 8, 2010

Abstract

Abstract

Pattern Matching is a very important process, which has its applications across different fields of scientific and non-scientific study. Since the task of pattern matching is computationally expensive and demanding, there is always a strong need to strive to develop algorithms which are more efficient than the ones presently in use. This paper aims at developing one such algorithm based on the concept of Fourier Transforms and its supplementary theorems to achieve a pattern match between specimen given as input to the algorithm.

This algorithm is supposed to be the core part of a much larger project in an important area in Medical Research that deals with matching Genes, cell samples efficiently and in a cost-effective manner to aid the fields like Genetic Engineering and Cancer Detection.

Outline

Notations Used

We use v_1, to denote the first set of data, v_2 to denote the second set of data. Let

$\mathbf{v_1} = a_1, a_2, \ldots, a_n$ and

$\mathbf{v_2} = b_1, b_2, \ldots, b_m$

Also $< \mathbf{v_1}, \mathbf{v_2} >$ indicates the dot product of $\mathbf{v_1}$ and $\mathbf{v_2}$.

$\rho_{\text{norm}(i)}$ - Normalized correlation coefficient between the vectors $\mathbf{v_1}$ and $\mathbf{v_2}$.

$\text{norm}(\mathbf{X}) = \sqrt{(\sum_{i=1}^{n} x_i^2)}$

$|\mathbf{X}| = \text{norm}(\mathbf{X})$

Definitions

Definition (**Flowgram**)

- A flowgram is a data file which consists of readings recorded by the Mass Spectrography Machine. These readings are the intensities of the light emitted by different element atoms in the specimen, when subjected to mass spectrography. Normally a flowgram consists of millions of data values.

Definition (**Correlation Coefficient**)

Correlation is any of a broad class of statistical relationships between two or more random variables or observed data values. For this paper, the Correlation Coefficient ρ calculated between two vectors v_1 and v_2 is given by,

$$\rho = \frac{< v_1, v_2 >}{|v_1||v_2|}$$

Definitions

Definition (**Fourier Transform**)

The Fourier transform (often abbreviated FT) is an operation that transforms one complex-valued function of a real variable into another. In such applications as signal processing, the domain of the original function is typically time and is accordingly called the time domain. The domain of the new function is typically called the frequency domain, and the new function itself is called the frequency domain representation of the original function. It describes which frequencies are present in the original function. The fourier transform of an integrable function $f : \mathrm{R} \rightarrow \mathrm{C}$ is given by,

$$F(k) = \int_{-\infty}^{\infty} f(x) e^{-2\pi i k x} dx \tag{2.1}$$

Here k is a real number

Definition (**Cross Correlation**)

The cross-correlation of two complex functions f(t) and g(t) of a real variable t, denoted by $f \star g$ is defined by,

$$f * g = \int_{-\infty}^{\infty} f(\bar{\tau}) g(t - \tau) d\tau \tag{2.2}$$

Definitions

Definition (**Kronecker Delta Function**)

The Kronecker Delta Function is a function of two variables, normally integers, which is 1 if they are equal, and 0 if they are not equal. i.e.

$$\begin{aligned} \delta_{1,2} &= 0 \\ \delta_{3,3} &= 1 \end{aligned}$$

It can be written as follows:

$$\delta_{ij} = 1, \text{if} i = j \tag{2.3}$$

$$\delta_{ij} = 0, \text{if} i \neq j \tag{2.4}$$

An important property of the Kronecker Delta Function is the **shifting property** i.e. for any $j \epsilon Z$

$$\sum_{i=-\infty}^{\infty} a_i \delta_{ij} = a_j \tag{2.5}$$

How the project was initiated? I

Pattern matching is a very important area of research in Computer Science, and deals with techniques which can be used to efficiently match two samples of data. The source of data can be anything ranging from pure mathematical functions to any empirical data obtained on conduction of an experiment. Two important areas of application of Pattern Matching are:

1. An area of medical research, which aims at detection of Cancer and other deadly diseases at an early stage. This can be done by matching the DNA of body cells with those of cancerous cells. Using the present techniques of pattern matching, it is a very inefficient and computationally intensive task and very expensive. Hence some new method is required, which can perform the task with efficiency in an affordable and time constrained manner.

How the project was initiated? II

2. In the field of Chemical Engineering, new chemicals are always being synthesized. The techniques for determining the molecular formula of a chemical mostly require repeated **mass spectrography** of the chemical compound. The data obtained from them are then fed to the **Pattern Matching Engine**, in order to obtain a consistent empirically derived molecular formula for the chemical/compound. This process monetarily very expensive and requires huge amount of computing resources. Moreover, the method is inefficient as it takes weeks and months to obtain accurate results.

Because of the shortcomings in the current methods employed for Pattern Matching, there was a strong necessity to devise a new and efficient method for the same.

Methods to Generate Source Data I

As outlined in the previous subsection, the current methods in order to achieve pattern matching involve **repeated Mass Spectrography** of the specimen compound or tissue. Every element atom (like C,H,Fe etc.), when subjected to mass spectrography (external stimuli) gives out light of a different wavelength. The Mass Spectrography Machine records the intensities of these emissions. Thus corresponding to every emission, there is a corresponding reading recorded by the Machine. All such readings are stored in a special file known as a **"Flowgram"**. Typically this file has millions of entries, corresponding to emissions by different element atoms in the specimen.

The process of generating a Flowgram requires one complete cycle run of the Mass Spectrometer Machine. On a single run, the Mass Spectrometer Machine records the emissions from a particular chain of atoms in the compound. The compound itself is made of **millions of such chains**. On every run, the Mass Spectrometer Machine records emissions from

Methods to Generate Source Data II

different chains of atoms in the compound, and generates the corresponding flowgram. Normally, thousands of such runs are made, and the corresponding flowgrams are stored.
Now is the real task at hand. The flowgrams may not represent contiguous chains of atoms in the compound, but may represent discrete chains or even overlapped chains of atoms in the compound. Hence, the task is to take all the flowgrams into consideration, find overlapped patterns in them, and to generate a single flowgram representing the molecular structure of the entire compound. Once this is done, repeating patterns can be further found in this single flowgram in order to estimate the molecular formula of the compound.
As is evident, the same process can be carried out for matching normal cell DNA patterns with cancerous cell DNA patterns,thus facilitating **the early detection of Cancer**.

The Traditional Method I

The traditional method to obtain pattern matching is perhaps the most obvious one. The following steps are involved in the traditional Method:

1. For the two vectors v_1 and v_2, first match the corresponding elements of v_1 and v_2 i.e. $a_1 \leftrightarrow b_1$, $a_2 \leftrightarrow b_2$ and so on upto $a_i \leftrightarrow b_i$. Here $i = \min(m, n)$
2. If a match is obtained then finish, else shift the vector v_2 one unit to the left and repeat step (1).

In scientific terms, using the concept of correlations of two vectors v_1 and v_2,

1. Calculate

$$\rho_{\text{norm1}} = \frac{< v_1, v_2 >}{\text{norm}(v_1)\text{norm}(v_2)}$$

The Traditional Method II

2. Then calculate

$$\rho_{\text{norm2}} = \frac{< v_1(1 : n-1), v_2(2 : n) >}{\text{norm}(v_1(1 : n-1))\text{norm}(v_2(2 : n))}$$

3. and so on....

The combination which gives the highest normalized coefficient ρ is the best match between the two vectors.
This approach to pattern matching, although simple to design and implement is very inefficient as is evident from its running times. For large vectors or Flowgrams in our case, the **time complexity** and the **cost of running** for this approach is found to be very high, and thus this method or approach is undesirable for large vectors. Presently, when this traditional approach is applied to Flowgrams, it takes several weeks to obtain the results.

Materials and Methods - Discrete Fourier Transform I

The discrete fourier transform is a special kind of Fourier Transform, which is used to transform a function in the time domain to the frequency domain. The main difference between the *discrete* and *continuous* variants of the Fourier Transform is that in case of the Discrete Fourier Transform, the input function should be discrete and their values should have a limited finite duration.
This is done by sampling a continuous function e.g. a person's voice. The DFT only evaluates that many frequency components to reconstruct the finite segment that was analyzed. Using the DFT, it is imperative that the finite segment analyzed be one period of an infinitely extended periodic signal. If this is not true, then a definite window has to be chosen to reduce make the function periodic within the concerned spectrum.
The input given to a DFT is a sequence of real of complex values. DFTs have several applications in the field of signal processing which make them a very important. Moreover, a key factor in favour of DFTs is that the

Materials and Methods - Discrete Fourier Transform II

DFT can be computed efficiently in practice using a **Fast Fourier Transform (FFT)** algorithm. This is very important from the performance point of view.

Materials and Methods - Cross-Correlation Theorem I

The cross-correlation is one of the most important theorems related to Disrete Fourier Transforms. The Cross-Correlation Theorem for both Continuous and Discrete functions can be stated as follows: *The cross-correlation of two functions f and g can be obtained as the inverse transform of the product of F^* and G, where F^* denotes the fourier transform of the conjugate of f, whereas G denotes the fourier transform of g*". i.e.

$$F^{-1}(F^*.G)_n = (f \star g_N)_n \tag{3.1}$$

$(f \star g_N)_n$ - denotes the correlation of f with a periodically extended g defined by ,

$$(g_N)_n = \sum_{p=-\infty}^{\infty} y(n - pN)$$

Let,

$f = [x_0 \qquad x_1 \ldots x_{N-1}]$

Materials and Methods - Cross-Correlation Theorem II

$g = [y_0 \quad y_1 \ldots y_{N-1}]$ The correlation of the two functions f and g is defined by,

$$(f \star g_N)_n = \sum_{m=0}^{N-1} x_m^* y(n+m) \tag{3.2}$$

$\star$ - The cross-correlation operator
F^{-1} - The inverse fourier transform operator

Proof I

Consider two discrete valued functions **f** and **g** with their corresponding fourier transforms as **F** and **G** respectively. **F*** is the fourier transform of the conjugate of f. **N** is the number of discrete points in **f** and **g**. **n** denotes the nth fourier transform.

Also $f = [x_0 \quad x_1 \ldots x_{N-1}]$

$g = [y_0 \quad y_1 \ldots y_{N-1}]$

$$\begin{aligned} \text{RHS} &= F^{-1}(F^*.G)_n \\ &= \frac{1}{N}\sum_{k=0}^{N-1} F_k^* Y_k e^{\frac{2\pi i}{N}kn} \qquad (3.3) \\ &= \frac{1}{N}\sum_{k=0}^{N-1}\left(\sum_{l=0}^{N-1} x_l^* e^{\frac{2\pi i}{N}kl}\right)\left(\sum_{m=0}^{N-1} y_m e^{-\frac{2\pi i}{N}km}\right) e^{\frac{2\pi i}{N}kn} \qquad (3.4) \\ &= \sum_{l=0}^{N-1} x_l^* \sum_{m=0}^{N-1} y_m \left(\frac{1}{N}\sum_{k=0}^{N-1} e^{\frac{2\pi ik}{N}(n+l-m)}\right) \qquad (3.5) \end{aligned}$$

Proof II

The term in the last equation above in the parentheses is 0 for all values of m except those of the form $n + l - pN$ where $p \epsilon Z$. At these values it is 1. Hence the sum in the parentheses can be replaced by the **Kronecker Delta Function** infinite sum. Also the limits of m can be extended to ∞, assuming that the f and g functions are defined as 0 outside [0,N-1] (**zero padding**). Thus we get,

$$F^{-1}(F^* G)_n = \sum_{l=0}^{N-1} x_l^* \sum_{m=-\infty}^{\infty} y_m \left(\sum_{p=-\infty}^{\infty} \delta_{m,(n+l-pN)} \right) \quad (3.6)$$

$$= \sum_{l=0}^{N-1} x_l^* \sum_{p=-\infty}^{\infty} \sum_{m=-\infty}^{\infty} y_m \delta_{m,(n+l-pN)} \quad (3.7)$$

Proof III

Now taking into account the **shifting property of the Kronecker Delta Function**, we have,

$$F^{-1}(F^* G)_n = \sum_{l=0}^{N-1} x_l^* \sum_{p=-\infty}^{\infty} y_{n+l-pN} \tag{3.8}$$

$$\tag{3.9}$$

Since f and g are assumed to be periodic, the factor pN will only cause the window to shift circularly, and hence would make no difference to the above equation. Thus

$$F^{-1}(F^* G)_n = \sum_{l=0}^{N-1} x_l^* y_{n+l} = (x \star y_N)_n \tag{3.10}$$

Thus, the correlation theorem is proved.

A Simple Example I

Let us start with a simple example in order to demonstrate the use of correlation coefficient in pattern matching. Let the discrete valued functions f and g be,

$$f = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \end{pmatrix}$$

$$g = \begin{pmatrix} 3 & 6 & 1 & 2 \end{pmatrix}$$

In this example, the back of g matches with the front of f. We now carry out the calculations of the correlation coefficients for all possible shifts of g with respect to f. After each step, we shift the g function one place to the left. The elements in consideration at each step are indicated by bolds. The correlation vector is denoted by ρ.

A Simple Example II

1. **No Shift**

$$f = (\mathbf{1}\ \mathbf{2}\ \mathbf{3}\ \mathbf{4})$$
$$g = (\mathbf{3}\ \mathbf{6}\ \mathbf{1}\ \mathbf{2})$$

$$\rho(1) = \frac{(1)(3)+(2)(6)+(3)(1)+(4)(2)}{norm([1\ \ 2\ \ 3\ \ 4]) * norm([3\ \ 6\ \ 1\ \ 2])} = 0.616$$

2. **Left Shift g by 1**

$$f = (\mathbf{1}\ \mathbf{2}\ \mathbf{3}\ 4\ 5)$$
$$g = (3\ \mathbf{6}\ \mathbf{1}\ \mathbf{2})$$

$$\rho(2) = \frac{(1)(6)+(2)(1)+(3)(2)}{norm([1\ \ 2\ \ 3]) * norm([6\ \ 1\ \ 2])} = 0.583$$

A Simple Example III

3. **Left Shift g by 2**

$$\begin{aligned} f &= \begin{pmatrix} \mathbf{1} & \mathbf{2} & 3 & 4 & 5 \end{pmatrix} \\ g &= \begin{pmatrix} 3 & 6 & \mathbf{1} & \mathbf{2} \end{pmatrix} \end{aligned}$$

$$\rho(3) = \frac{(1)(1) + (2)(2)}{norm([1 \quad 2]) * norm([1 \quad 2])} = 1$$

4. **Left Shift g by 3**

$$\begin{aligned} f &= \begin{pmatrix} \mathbf{1} & 2 & 3 & 4 & 5 \end{pmatrix} \\ g &= \begin{pmatrix} 3 & 6 & 1 & \mathbf{2} \end{pmatrix} \end{aligned}$$

$$\rho(4) = \frac{(1)(2)}{norm([1]) * norm([2])} = 1$$

A Simple Example IV

It should be noted that for an overlap of a single element, the value of ρ will always be 1 and hence it should not be considered while determining the maximum value of ρ. Hence, excluding the last case, wherein the overlap was of a single element, the value of ρ is greatest for case no. 3 with an overlap of 2 elements. Hence, it is the best match that can be found given these two vectors.
Hence the complete joined vector **h** can be represented as:

$$h = \text{join(f,g)} = \begin{pmatrix} \mathbf{3} & \mathbf{6} & \mathbf{1} & \mathbf{2} & \mathbf{3} & \mathbf{4} & \mathbf{5} \end{pmatrix}$$

How the theorem is applied?

As seen above from the cross-correlation theorem, we get a way to compute the correlation between two given integrable functions $f(t)$ and $g(t)$.
Pattern Matching always requires some similarity measure between the functions under consideration to gain a quantitative estimation of just how similar or dissimilar two functions are. Deviating away from the traditional approach here, we use the Cross-Correlation theorem in order to compute the Correlation between the functions (obtained from the source flowgrams as discrete valued functions) under consideration instead of the traditional compare and shift approach. It is very important to note that in this case **we assume the flowgrams to be of equal length**.

Zero Padding and its requirement

As it is clear from the the correlation theorem, its proof assumes that both the functions f and g are periodic in nature with period N. In case of discrete values, this may not always be the case. Hence in order to successfully apply the correlation theorem to such type of functions, we need to add periodicity to non-periodic functions.
i.e. the function is assumed to repeat the same set of values of different windows of size N with no overlappings among themselves. We choose only one of the window of the N elements. The only way to add periodicity to these type of functions is to introduce or pad zeros to either the front or back one of the functions (having smaller length). Doing so serves both the purposes, i.e. it makes the window size definite for both functions (periodicity introduced) and introduction of zeros does not affect the calculation of the correlation coefficients in any way.
Thus it is imperative to accomplish zero padding prior to applying the correlation theorem to functions representing sequences of unequal length in order to obtain a successful correlation coefficient vector for different function shifts.

Interpreting the Results

Using the **cross-correlation theorem**, we can get the correlation between the two flowgrams using fourier transforms. Doing so saves a lot of computation time, and gives the result in a single vector wherein each component represents the correlation coefficient for a particular offset. We can then find the maximum value of the correlation coefficient corresponding to an offset and thus obtain **the maximum similarity** that exists between the two flowgrams. If the level of similarity obtained is above a certain threshold, the two flowgrams can be combined into a single flowgram. In effect we are joining separated portions of a chain of atoms. Repeating the above process iteratively reduces the joins flowgrams in each iteration if the match is above the threshold. Having an upper limit on the maximum number of iterations, we can make the process deterministic in nature.

The final output of this process is a combined flowgram (along with certain unmatched flowgrams) which represents a chain of representing the molecular structure of the specimen compound. In effect, this process makes the largest chain possible by matching patterns in different source flowgrams.

A Note on Demeaning I

The process of **demeaning** a vector is necessary in order to make the average value of all the elements in the vector equal to zero. If this process is not applied to the Flowgrams(vectors) beforehand, then it causes random noise to be added in the results, thus reducing their accuracy. We now outline the formal algorithm to achieve the same. There are two algorithms,

1. The first algorithm deals with pure computational functionality. i.e. It just computes the cross-correlation between two vectors given using the fast-fourier Transforms.
2. The second algorithm performs pattern matching on all possible combinations between the two vectors which includes the following: Given two vectors f_1 and f_2,
    1. The back of f_1 matches with the front of f_2 and vice versa
    2. The front of f_1 matches with the back of f_2 and vice versa
    3. f_2 is entirely a part of f_1 and vice versa

Algorithm to compute the cross-correlation between two given Flowgrams

Require: $f_1 \leftarrow$ Flowgram 1 (an m column vector)
$f_2 \leftarrow$ Flowgram 2 (an n column vector)
$m >= n$
$\lambda \leftarrow$ Threshold on the minimum no. elements that must match between two flowgrams to join them ρ_{thresh} Threshold on the minimum value of ρ required to conclude a successful match

1: A $\leftarrow$ ifft(conj(fft(f_1)*fft(f_2))) {Calculate numerator using FFT}
2: $e \leftarrow$ ONES(length(f_2)) {Define a vector of all ones of length n}
3: B $\leftarrow$ ifft(conj(fft(f_1)*fft(e))) {Calculate norm of f_1 terms}
4: C $\leftarrow$ ifft(conj(fft(f_2)*fft(e))) {Calculate norm of f_2 terms}
5: $\rho_{norm} \leftarrow \frac{A}{\sqrt{B}\sqrt{C}}$
6: return ρ_{norm}

Algorithm to achieve a Pattern Match between two Flowgrams I

Require: $f_1 \leftarrow$ Flowgram 1 (an m column vector)
$f_2 \leftarrow$ Flowgram 2 (an n column vector)
$m >= n$
$\lambda \leftarrow$ Threshold on the minimum no. elements that must match between two flowgrams to join them ρ_{thresh} Threshold on the minimum value of ρ required to conclude a successful match

1: {No zero padding required}
2: **if** length(f_1)==length(f_2) **then**
3: $\rho_{res1} \leftarrow FOURIERMATCH(f_1, f_2, \rho_{thresh}, \lambda)$ {Match back of f_1 with front of f_2}
4: $\rho_{res2} \leftarrow FOURIERMATCH(f_2, f_1, \rho_{thresh}, \lambda)$ {Match front of f_1 with back of f_2}
5: $(m_1, i_1) = max(\rho_{res1}(1 : end - 1))$ {Get best match offset and rho value for first match attempt}

Algorithm to achieve a Pattern Match between two Flowgrams II

6: $(m_2, i_2) = max(\rho_{res2}(1 : end - 1))$ {Get best match offset and rho value for second match attempt}
7: Choose appropriate pair m_i, i_i depending upon the values of ρ_{thresh} and threshold respectively
8: **else if** length(f_1)>length(f_2) **then**
9: {Zero padding required here}
$l \leftarrow$ length(f_1) - length(f_2)
$f_2 \leftarrow$ [f_2;zeros(len,1)] {Pad zeros onto the front of f_2}
10: $\rho_{res1} \leftarrow FOURIERMATCH(f_2, f_1, \rho_{thresh}, \lambda)$
11: $f_2 =$ [zeros(len,1);f_2] {Pad zeros onto the back of f_2}
12: $\rho_{res2} \leftarrow FOURIERMATCH(f_1, f_2, \rho_{thresh}, \lambda)$
13: $(m_1, i_1) = max(\rho_{res1}(1 : end - 1))$ {Get best match offset and rho value for first match attempt}

Algorithm to achieve a Pattern Match between two Flowgrams III

14: $(m_2, i_2) = max(\rho_{res2}(1 : end - 1))$ {Get best match offset and rho value for second match attempt}

15: Choose appropriate pair m_i, i_i depending upon the values of ρ_{thresh} and threshold respectively

16: **end if**

Results I

This section give the results obtained on actual implementation of the pattern matching technique outlined above. We also make a head-to-head comparison between the standard approach and the fourier-transform based approach to test the efficiency of this method of pattern matching:
We randomly initialize $\mathbf{f_1}$ and $\mathbf{f_2}$, the two source flowgrams such that length($\mathbf{f_1}$)<=length($\mathbf{f_2}$).
Moreover, we intentionally make the back of f_2 match with the front of f_1 (with some added noise in order to simulate a real world situation). The figures on the subsequent pages show three graphs:

1. **Graph 1** - This graph represents the value of the correlation coefficients corresponding to different values of the offsets of f_2 with respect to f_1. Also the graph shows the maximum value of the correlation coefficient obtained for any possible offset. Note that the value of the correlation coefficient for an overlap of a single element

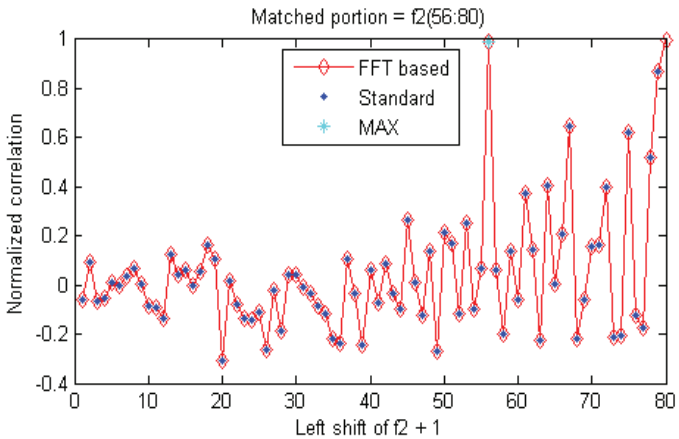
Results II

between the two functions is always 1 and hence is not considered while pattern matching.

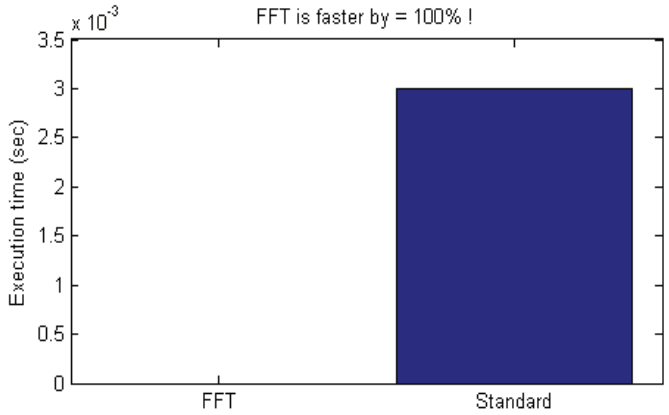
2. **Graph 2** - This is the time graph, which shows a head to head performance of the normal method and the fourier transform based method. As it is evident for all the cases, the fourier transform method clearly outperforms the traditional method, and is a major improvement over the traditional method.
3. **Graph 3** - This is the graph representing the two functions f and g along with the shifted g function and the final joined function. This gives an idea of how the joining takes place and how the joined vector looks like.

Output I

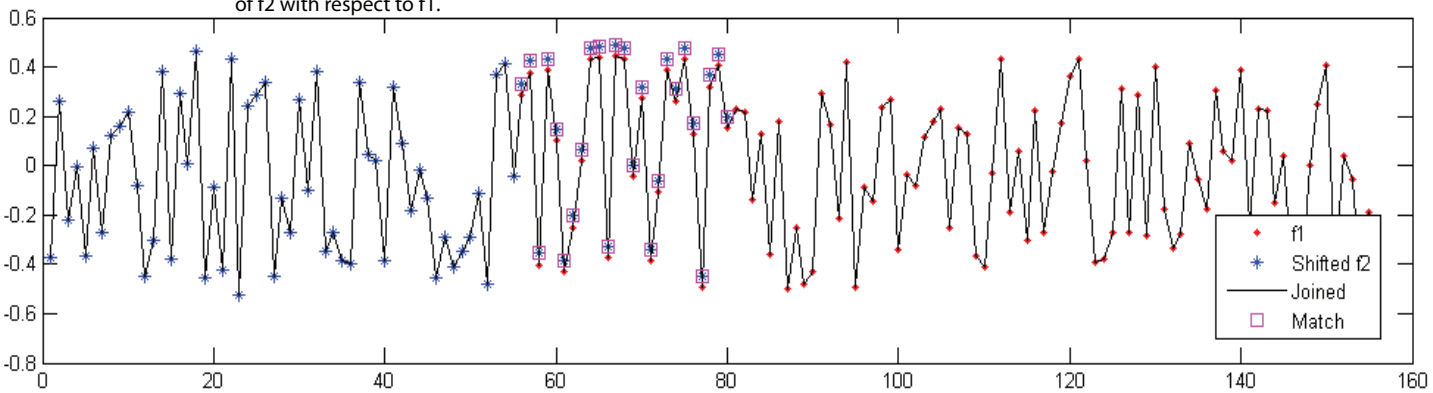
Results obtained for a test run of the algorithm for random initial values of f1 and f2 respectively. Here f1 = rand(100,1) while f2 = rand(80,1). We intentionally make the back of f2 match with the front of f1 (with some added noise of course) in order to verify the correctness of the process.



The normalized corelation obtained for different shifts of f2 with respect to f1.Each value indicates the cross-correlation for a particular shift of f2 with respect to f1.



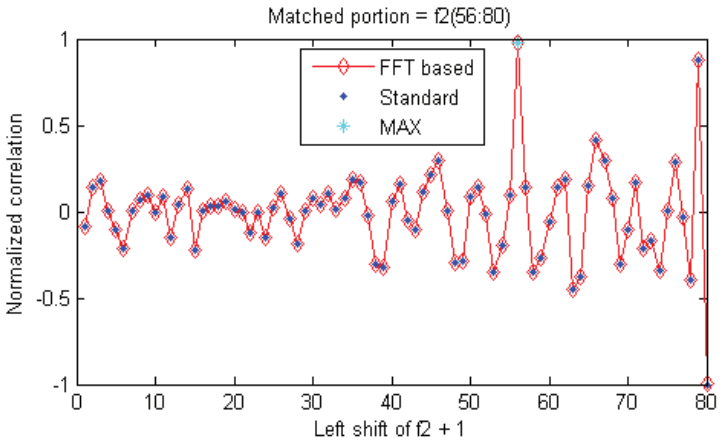
The running times for the standard approach and the fourier transform based approach. It is clear, that the FFT based approach is much faster than the traditional one



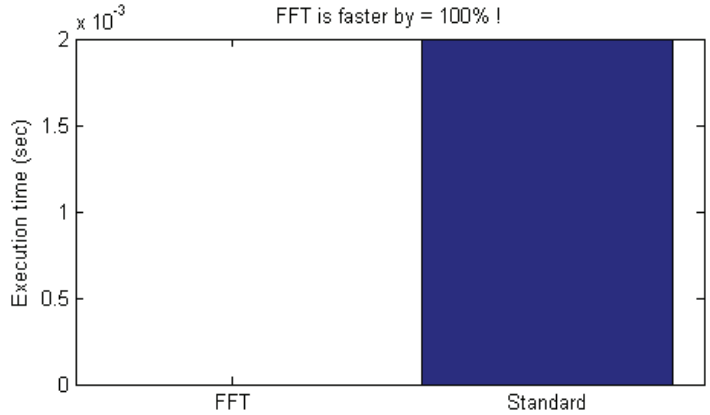
This graph gives the combined flowgram, which is a combination of f1 followed by f2. The legends in the graph show the corresponding flowgrams, the joined flowgram and the portion where the match is found between the two flowgrams i.e. the overlapping portion between the two flowgrams

Output II

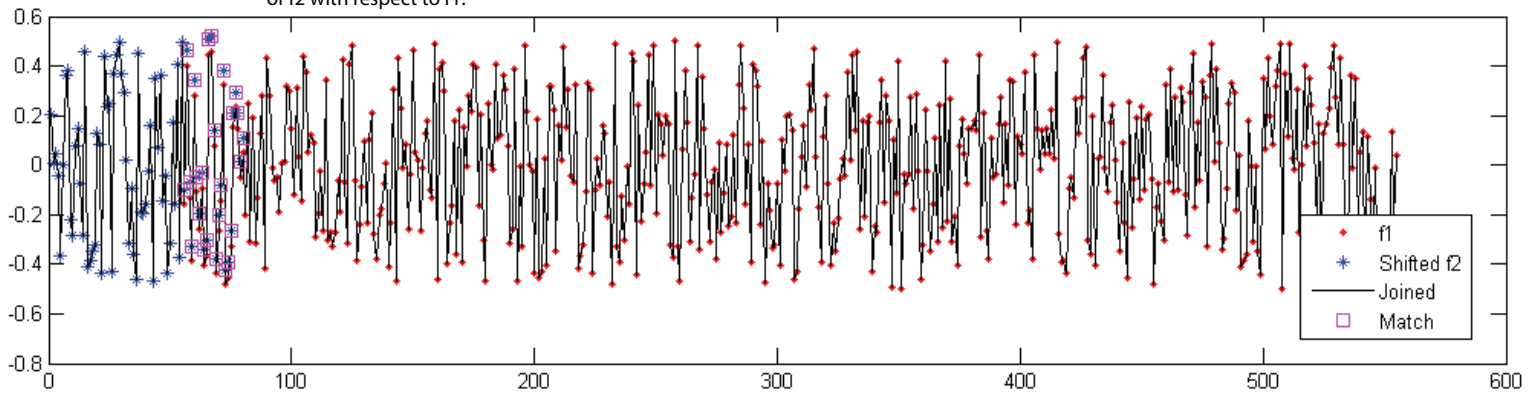
Results obtained for a test run of the algorithm for random initial values of f1 and f2 respectively. Here f1 = rand(500,1) while f2 = rand(80,1). We intentionally make the back of f2 match with the front of f1 (with some added noise of course) in order to verify the correctness of the process.



The normalized corelation obtained for different shifts of f2 with respect to f1. Each value indicates the cross-correlation for a particular shift of f2 with respect to f1.

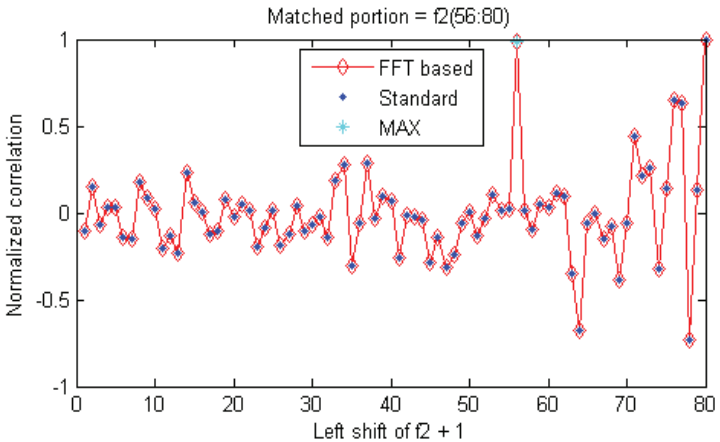


The running times for the standard approach and the fourier transform based approach. It is clear, that the FFT based approach is much faster than the traditional one

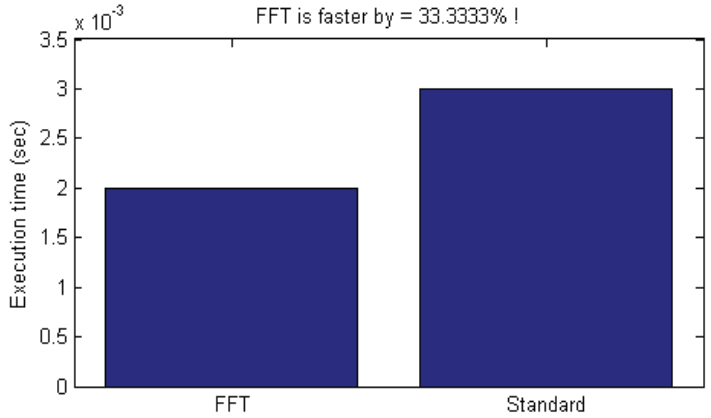


Output III

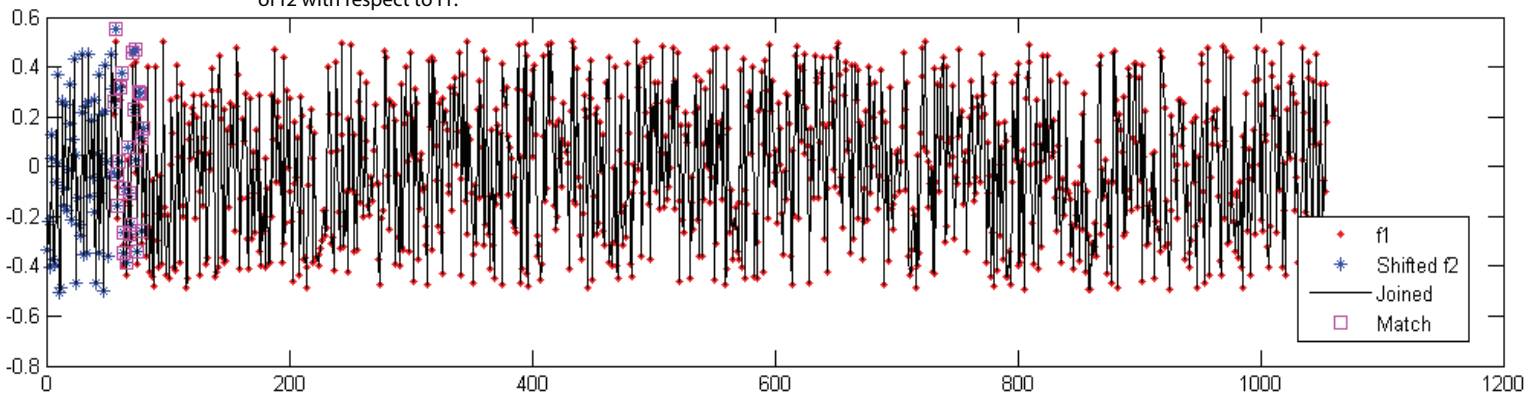
Results obtained for a test run of the algorithm for random initial values of f1 and f2 respectively. Here f1 = rand(1000,1) while f2 = rand(80,1). We intentionally make the back of f2 match with the front of f1 (with some added noise of course) in order to verify the correctness of the process.



The normalized corelation obtained for different shifts of f2 with respect to f1. Each value indicates the cross-correlation for a particular shift of f2 with respect to f1.



The running times for the standard approach and the fourier transform based approach. In this case, the FFT based method is 33.33% faster than the traditional approach.



This graph gives the combined flowgram, which is a combination of f1 followed by f2. The legends in the graph show the corresponding flowgrams, the joined flowgram and the portion where the match is found between the two flowgrams i.e. the overlapping portion between the two flowgrams.

Discussion

From the above actual runs of the pattern matching algorithm, for randomly generated flowgrams f_1 and f_2, we can say that the FFT based approach is considerably faster than the traditional approach. Thus it is safe to postulate that the FFT based approach saves a lot of computation time when matching thousands of flowgrams. Moreover, the user can set the thresholds on the value of the cross-correlation ρ, the minimum number of elements which are required to match in order to join the flowgrams, and the maximum number of iterations to be performed when matching more than two flowgrams.

Conclusion

Thus we can conclude that this new approach to pattern matching using the concepts of Fourier Transforms and the Cross-Correlation or Convolution theorem is much superior to the traditional approach to pattern matching. Actual implementation of this approach on a target system will definitely improve its performance manifolds.