[SPIE, Vol. 2615, pp 29-39, 1995]

Image Indexing And Retrieval Using Linear Phase Coefficient Composite Filters

Mark J. Carlotto

PSR Corporation, 1400 Key Blvd., Suite 700, Arlington, VA 22209
(mjcarlotto@ma.psrw.com)

Abstract

Content-based retrieval techniques can be characterized in several ways: by the manner in which image data are indexed, by the level of specificity/generality of the query and response of the system, by the type of query (e.g., iconic or symbolic), and by the kind of information used (intrinsic image features or attached information such as text). The method described in this paper automatically indexes images in the database, and is intended to retrieve specific objects by image query based on inherent image content. Our method is actually quite similar to object recognition except that instead of searching a single image for a given object, an entire database of images is examined. The approach uses linear phase coefficient composite (LPCC) filters to encode and match queries consisting of multiple images (e.g., representative views of an object of interest) against multiple images in the database simultaneously. Retrieval is a two-step process that first isolates those portions of the database containing images that match the query, and then identifies the specific images. Our use of LPCC filters exploits phase information to retrieve specific images that match the query from the database. The results from the experiments suggest that phase information can be used to index and retrieve multiple images from a database in parallel, and that large numbers of operations can be performed simultaneously using a complex number representation. In one experiment well over 100 real correlations were effectively performed by a single complex correlation. Problems encountered in processing video data are discussed.

Keywords: content-based image retrieval, linear phase coefficient composite filters, phase coding

1. Introduction

Content-based retrieval techniques can be characterized in several ways: by the manner in which image data are indexed, by the level of specificity/generality of the query and response of the system, by the type of query (e.g., iconic or symbolic), and by the kind of information used (intrinsic image features or attached information such as text). For example, some systems require manual or interactive indexing to outline potential objects of interest, textures, etc., while others extract features of the image as a whole or of objects within the image that can be used as indices. The level of specificity/generality of the system is a function of the underlying representations. For example color histograms allow images similar in the global distribution of colors to be found given a query image but cannot identify images containing specific objects.

Representation of shape features allows images to be retrieved that contain specific objects that match the shape of the query, subject to a set of allowable distortions. Query by example systems require some form of iconic input (either a graphical sketch or image example). More traditional data retrieval systems use symbolic input (e.g., text, SQL, or other descriptions of the objects/images of interest. Finally some image retrieval systems do not actually use intrinsic features of the images themselves but rather attached information in the form of relational or object-oriented description or free-text. (See [1] for an early discussion of alternative access methods for image retrieval. The system described in [2] embodies many of the above ideas and is among the most mature content-based image retrieval today. Descriptions of other techniques and systems developed to date can be found in [3-5].)

According to the above criteria, the method described in this paper automatically indexes images in the database, and is intended to retrieve specific objects by image query based on inherent image content. Our method is actually quite similar to object recognition except that instead of searching a single image for a given object, an entire database of images is examined. The approach uses linear phase coefficient composite (LPCC) filters [6] to encode and match queries consisting of multiple images (e.g., representative views of an object of interest) against multiple images in the database simultaneously. Retrieval is a two-step process that first isolates those portions of the database containing images that match the query, and then identifies the specific images. Our use of LPCC filters exploits phase information to retrieve specific images from the database that match the query.

The next section reviews synthetic discriminant function (SDF), and LPCC filters. Application of LPCC filters to image indexing and retrieval are discussed in Section 3. Differences between our LPCC filter implementation and that of Hassebrook et al [6] are noted. Results from several experiments performed on a database of images captured from a news broadcast are presented in Section 4. Section 5 discusses advantages and disadvantages of LPCC filters for content-based retrieval.

2. Synthetic Discriminant Function Filters

Two basic approaches in object recognition are to either characterize and recognize objects of interest in terms of features that are invariant to the kinds of distortions expected, or to construct and match a set of representative views of the object to the image. SDF filters are an example of the later approach. If $\mathbf{H} = \{\mathbf{h}_k\}$ is a matrix whose columns contain the representative views (training images) arranged in lexographic order, the general form of the SDF filter is

$$\mathbf{f} = \mathbf{H}\mathbf{w} \tag{1}$$

where the weight vector $\mathbf{w}$ is computed to achieve a desired response.

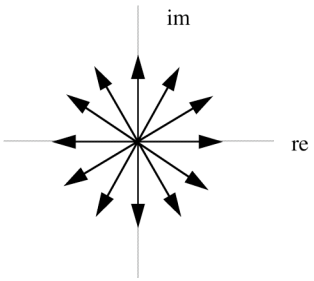

Fig. 1 LPCC weights are uniformly spaced phasors

The LPCC filter is a kind of SDF filter where the weight vector consists of linear phase coefficients [7]. The general form of the n-th order LPCC filter is:

$$\mathbf{f}_n = \mathbf{H}\mathbf{w}_n, \text{ or } \mathbf{f}_n = \sum_k \mathbf{h}_k e^{j2\pi kn/K}. \tag{2}$$

Hassebrook et al have used multiple LPCC filters both to improve filter SNR as well as for parameter estimation. Consider the first-order filter

$$\mathbf{f} = \sum_k \mathbf{h}_k e^{j2\pi k/K} \tag{3}$$

where the weights are uniformly-spaced phasors as shown in Fig. 1. Let $z = \mathbf{f}^T\mathbf{g}$ be the output of the filter in response to the input $\mathbf{g}$. Assume the training images are real, non-negative, and normalized to unit energy; i.e.,

$$\mathbf{h}_k^T \mathbf{h}_k = 1. \tag{4}$$

If the input consists of one of the training images $\mathbf{h}_{k'}$ plus additive white Gaussian noise $\mathbf{n}$, the expected value of the response of the LPCC filter is

$$E[z] = E\left[\left(\sum_k \mathbf{h}_k e^{j2\pi k/K}\right)^T \left(\mathbf{h}_{k'} + \mathbf{n}\right)\right] = e^{j2\pi k'/K}. \tag{5}$$

For noise alone, the expected value is zero. The responses of the LPCC filter to one of the training images and to noise are depicted in Fig. 2. The signal-to-noise ratio at the output of the filter

$$SNR = \lambda d / K\sigma^2 \tag{6}$$

where σ^2 is the variance of the noise, λ is a factor that depends on the cross-correlation between training images, d is the size of (number of pixels in) the training images, and K is the number of training images stored in the filter [6].

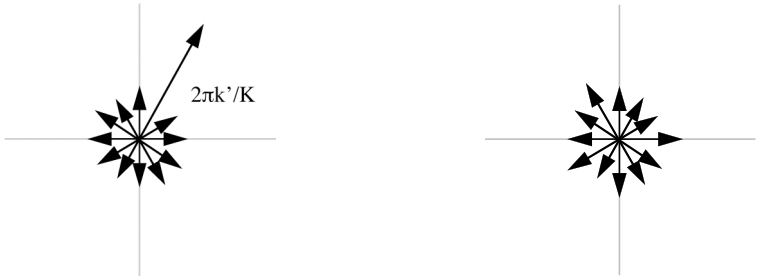

Fig. 2 Graphical depiction of LPCC filter response to a training image (left) and noise (right)

3. Application Of LPCCFs To Image Indexing And Retrieval

We group images in the database into segments and use LPCC filters to encode and represent all of the images within a segment in a single filter. LPCC filters are also used to encode multiple images in a query as is done in object detection. Fig. 3 provides an overview of the image indexing and retrieval process. The database is partitioned into segments and phase-encoded off line. The p-th segment of the database consisting of images $\mathbf{h}_{p,1}$ through $\mathbf{h}_{p,K}$ are phase encoded and stored in the frequency domain as $\mathbf{F}_W(p)$. At run time, a query is constructed, phase encoded if more than one image is involved, and correlated against the P phase encoded database segments. The output of the matching process consists of P correlation surfaces. The retrieval algorithm first identifies those segments that are likely to contain images that match the query based on the magnitude of the correlation, and then identifies specific images within the segment using the phase. The following sub-sections describe the phase encoding, complex correlation, and image retrieval processes in greater detail.

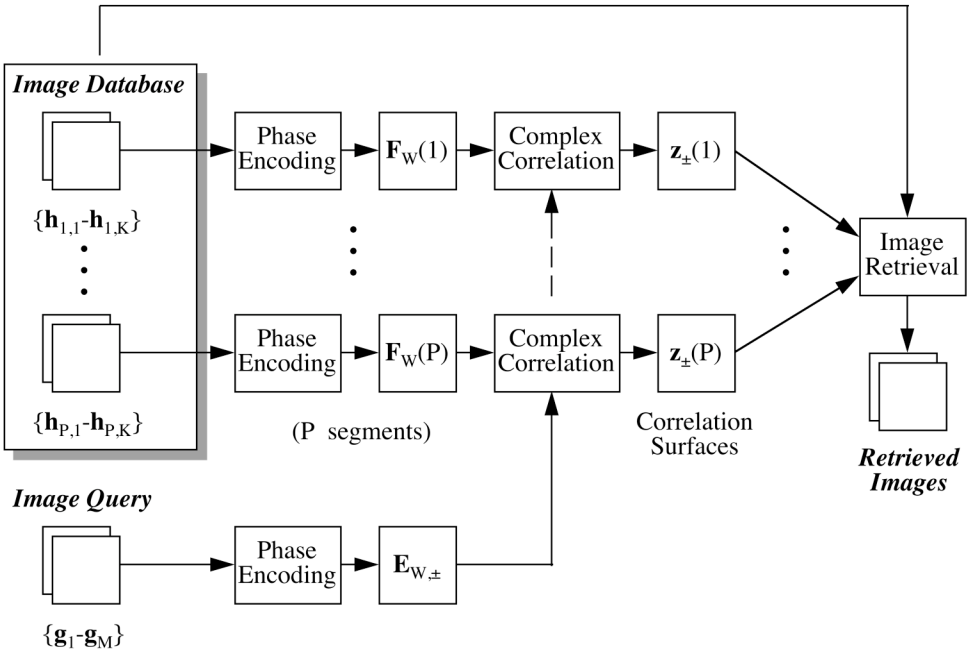

Fig. 3 Overview of the image indexing and retrieval process

3.1 Phase Encoding

Hassebrook et al edge enhance and normalize the energy of the images prior to phase encoding [6]. Instead of edge enhancing we whiten the images to maximize the SNR. For efficiency, correlation is performed in the frequency domain using FFTs. Phase encoding is performed first, followed by transformation to the frequency domain where whitening is performed before filtering. As shown below, this sequence of operations is equivalent to whitening, phase encoding, and transforming provided that the whitening operation $\mathbf{W}$ is linear. For database images $\{\mathbf{h}_k\}$ within a particular segment :

$$\begin{aligned}\Im^{-1}\left[\mathbf{F}_W\right] &= \Im^{-1}\left[\mathbf{W}\,\Im\left[\sum_k \mathbf{h}_k e^{j2\pi k/K}\right]\right] = \Im^{-1}\left[\mathbf{W}\sum_k \mathbf{H}_k e^{j2\pi k/K}\right] \\ &= \Im^{-1}\left[\sum_k \mathbf{H}_k^W e^{j2\pi k/K}\right] = \sum_k \mathbf{h}_k^W e^{j2\pi k/K}.\end{aligned} \tag{7}$$

Whitening is performed by setting the magnitude of the DFT to a constant without altering the phase [8]. It is noted that since this is a non-linear operation, Eq. 7 is not strictly valid. Queries are encoded in a similar manner. Those containing multiple images $\{\mathbf{g}_m\}$ are encoded by two sets of linear phase coefficients and stored as $\mathbf{E}_{W,\pm}$ where

$$\begin{aligned}\Im^{-1}\left[\mathbf{E}_{W,\pm}\right] &= \Im^{-1}\left[\mathbf{W}\,\Im\left[\sum_m \mathbf{g}_m e^{\pm j2\pi n/M}\right]\right] = \Im^{-1}\left[\mathbf{W}\sum_m \mathbf{G}_m e^{\pm j2\pi m/M}\right] \\ &= \Im^{-1}\left[\sum_m \mathbf{G}_m^W e^{\pm j2\pi m/M}\right] = \sum_m \mathbf{g}_m^W e^{\pm j2\pi n/M}\end{aligned} \tag{8}$$

3.2 Complex Correlation

Complex correlation is performed in the frequency domain:

$$\mathbf{z}_\pm = \{z_\pm(x,y)\} = \Im^{-1}\left[\mathbf{F}_W^* \mathbf{E}_{W,\pm}\right] = \left(\sum_k \mathbf{h}_k^W e^{j2\pi k/K}\right) * \left(\sum_m \mathbf{g}_m^W e^{\pm j2\pi m/M}\right) \tag{9}$$

where * denotes correlation. The location of the correlation peak, magnitude squared of the correlation peak, and phase of the correlation peak are obtained from the correlation surface:

$$(x',y') = \arg\max_{(x,y)}\left\{z_+(x,y)z_+^*(x,y)\right\} \tag{10a}$$

$$z' = z_+(x', y')z_+^*(x', y') \tag{10b}$$

$$\phi_\pm = \tan^{-1}\left(\mathrm{Im}\left[z_\pm(x', y')\right] / \mathrm{Re}\left[z_\pm(x', y')\right]\right) \tag{10c}$$

In object recognition, the position of the correlation peak locates possible object instances in the image. Here we are more interested in the magnitude of the correlation peak and the phase at the peak location.

3.3 Image Retrieval

Initially the magnitude of the correlation peak is used to rank order the segments; i.e., those with large correlation peak values are more likely to contain images that match the query than those with lower values. Having identified the segments of the database most likely to contain images of interest, the phase is used to identify specific images within the segment. If the query contains a single image, the phase obtained from $\mathbf{z}_+$ is used as an estimate of the image index

$$\hat{k} = K(\phi_+ / 2\pi). \tag{11}$$

For queries that contain multiple images, the image index is estimated from the phases of $\mathbf{z}_+$ and $\mathbf{z}_-$ at the peak location:

$$\hat{k} = K(\phi_+ + \phi_-)/4\pi, \left(K/2 + K(\phi_+ + \phi_-)/4\pi\right) \bmod K. \tag{12}$$

The ambiguity of K/2 results from the averaging of the phase angles as shown in the appendix.

4. Experimental Results

Images from a 30 minute segment of CNN's Headline News were used to evaluate our method. One video frame every 10 seconds was captured (163 frames total). The frames were then reduced to 260x180 pixel images. Thumbnails of all 163 frames are shown in Fig. 4.

Fig. 4 Image database derived from CNN Headline News broadcast

4.1 Comparison Of Implementations

First we constructed an LPCC filter from the first 20 frames of the video by our implementation (Eq. 7). As a test, a query image (16x16 pixels) was extracted from frame 2 (Fig. 5a) and correlated with the filter. The position of the correlation peak was at the location of the query object. The phase at the peak location was 13¡ which is close to the true value of the encode phase for frame 2, 360¡ x 2/20 = 18¡.

Next, we repeated the experiment using an LPCC filter implemented by edge enhancing and normalizing the database images prior to phase encoding as described by Hassebrook et al. The position of the correlation peak was again at the location of the query object. The measured phase was 16¡, slightly closer to the true value.

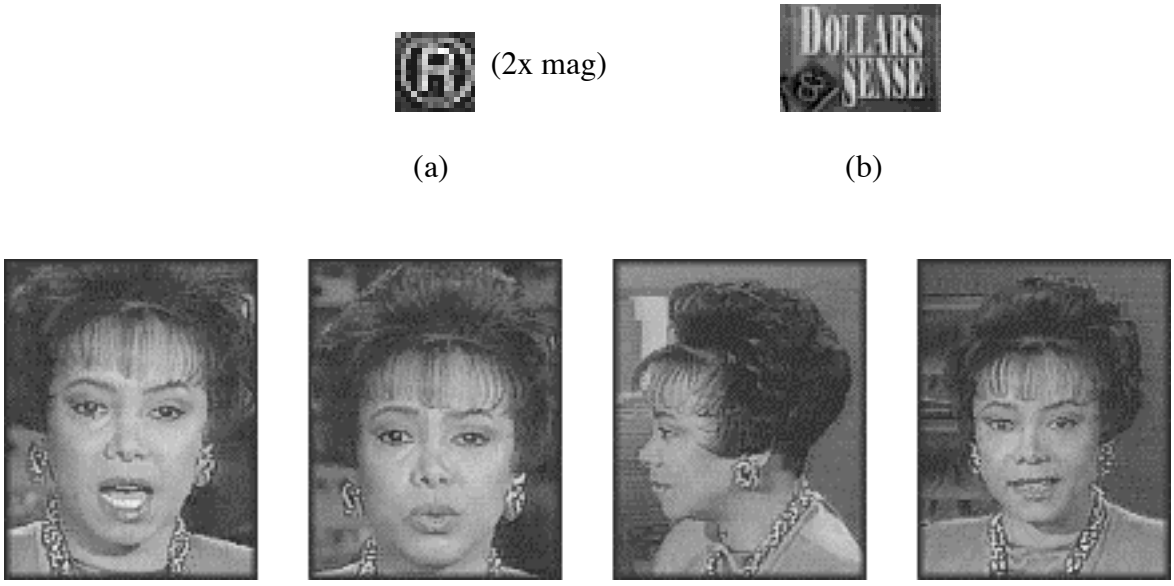

Fig. 5 Queries used in image retrieval experiments

We then partitioned the database into 8 segments of 20 images each, phase encoded each segment using both LPCC filter implementations and correlated the query image with the two sets of LPCC filters. The peak response and rank of the segments based on the peak responses for Hassebrook's implementation were:

1-20	21-40	41-60	61-80	81-100	101-120	121-140	141-163
3022 (6)	3304 (5)	6037 (3)	4066 (4)	6999 (1)	2423 (8)	6170 (2)	2781 (7)

The segment containing the query was ranked sixth. The same results using our LPCC filter implementation were:

1-20	21-40	41-60	61-80	81-100	101-120	121-140	141-163
7327 (2)	6510 (3)	12348 (1)	6443 (4)	5857 (6)	4666 (8)	6259 (5)	5203 (7)

with the segment containing the query ranked second. We found that the Headline News logo in the lower right corner of frames 57-60 in segment 41-60 correlated strongly with the query. The position of the frames within the segment were then changed so that their encoded phases were 90¡ apart. After redistributing these frames, the peak value for the segment decreased from 12348 to 5382. In the resultant ranking, segment 1-20 was ranked the highest as shown below.

1-20	21-40	41-60	61-80	81-100	101-120	121-140	141-163
7327 (1)	6510 (2)	5382 (6)	6443 (3)	5857 (5)	4666 (8)	6259 (4)	5203 (7)

4.2 Performance For Different Segment Sizes

The performance of our LPCC filter implementation was then evaluated over the database for segment sizes of 20, 40, 80, and 163 images per segment. In these experiments a larger query 60x40 pixels in size was used (Fig. 5b.) This pattern was present in frames 91 and 112 in the same location. The results for 8 segments containing 20 images were:

1-20	21-40	41-60	61-80	81-100	101-120	121-140	141-163
4046 (8)	6495 (4)	8116 (3)	4794 (6)	8609 (2)	10782 (1)	5880 (5)	4713 (7)

Segments 81-100 and 101-120 were ranked the highest. The measured phases for these two segments, 182¡ and 190¡, are close to the true phases for frames 91 and 112, 360¡ x (91-80)/20 = 180¡ and 360¡ x (112-101)/20 = 198¡, respectively.

The experiment was repeated for 4 segments containing 40 images each. The results show segment 81-120 which contains both frames 91 and 112 had the lowest score:

1-40	41-80	81-120	121-163
4713 (3)	5345 (1)	4068 (4)	5241 (2)

Since the phases assigned to frames 91 and 112 in segment 81-120 were almost 180¡ apart and both matched the query image in the same place, the individual responses canceled. The net response was thus the lowest. After removing frame 91 from the database and recoding, the resultant responses were

1-40	41-80	81-120	121-163
4713 (4)	5345 (2)	6519 (1)	5241 (3)

The measured phase for segment 81-120 was 263¡ which is close to the true phase of frame 112 (frame 91 eliminated), 360¡ x (112-81-1)/(40-1) = 277¡.

We then constructed LPCC filters for segments 1-80 and 81-163 (frame 91 was put back into the database). The responses were 3906 and 6903, respectively. The measured phase for segment 81-163 was 84¡ which was close to the average of the true phases of frames 91 and 112, 88.5¡. When two matches occur in the same place, the estimated index lies in between the true image indices.

Finally all 163 images were encoded in a single LPCC filter. The correlation peak occurred at the correct location with a measured phase of 216¡. The actual phases of frames 91 and 112 were 200¡ and 247¡, their average was 224¡.

4.3 Queries Containing Multiple Images

In this last set of experiments we evaluated the performance of the method for queries containing multiple images. The database was partitioned into 8 segments of 20 images each. Four facial images (95x120 pixels) at locations (206,246), (259,229), (327,237) and (334,236) from frames 9, 59, 112, and 162 (Fig. 5c) were phase encoded and stored in a LPCC filter which was correlated against the 8 LPCC filters corresponding to segments 1-20, 21-40, etc. The results below summarize the peak value of the response as before for all segments

1-20	21-40	41-60	61-80	81-100	101-120	121-140	141-163
4624 @ (206,246) $\phi_+ = 331¡$ $\phi_- = 274¡$ $\phi_k =$ 122¡, 302¡ (144¡)	3623	4439 @ (259,229) $\phi_+ = 238¡$ $\phi_- = 65¡$ $\phi_k =$ 151¡, 331¡ (324¡)	3512	3815	4618 @ (327,237) $\phi_+ = 34¡$ $\phi_- = 7¡$ $\phi_k =$ 20¡, 200¡ (198¡)	3525	5732 @ (334,236) $\phi_+ = 44¡$ $\phi_- = 213¡$ $\phi_k =$ 128¡, 308¡ (328¡)

For the four segments containing one of the query images we also show the location of the peak response, the phases measured from the $\mathbf{z}_+$ and $\mathbf{z}_-$ correlation surfaces, and the estimated phases, and the actual encoded phase of the image in the database matching the query at that location. All segments matching the query image were correctly identified and the phase estimated to within 13¡ of the true phase on average (±180¡).

5. Conclusion

The results from the experiments suggest that phase information can be used to index and retrieve multiple images in a database in parallel, and that large numbers of operations can be performed

simultaneously using a complex number representation. In one experiment 163 real correlations were effectively performed by a single complex correlation.

We found that the method is not without problems however. Objects of interest that occur in the same location in different images in the database will not be found if the corresponding images are assigned phases 180¡ apart. If the assigned phases are not 180¡ and if the correlation is large enough, the measured phase at the peak location will lie in between the assigned phases. We also found patterns that only weakly correlate with the query may strongly correlate and produce false positives if they occur in the same location and are assigned similar phases. Many of these problems are particularly relevant for video as we have seen but may be less of a problem for digitized photography.

Although we have shown that it is possible to match queries containing multiple images against segments containing multiple images, there is an ambiguity of 180¡ in the measured phase. In these cases it will thus take twice as long to find the correct image within a segment since, on average, twice as many images will need to be retrieved.

In cases where more than one image in a query match images in a segment, multiple peaks in the correlation surface must be examined. The phase at each peak location ±180¡ is used to locate candidate images within the segment that match one of the query images. This is an area of future work.

As shown previously in target detection, LPCC filters demonstrate the feasibility of matching a large number of representative views of the object of interest to images. Similar approaches may thus be appropriate in certain types of image retrieval problems that involve the recognition of particular faces, icons (e.g., to detect key frames), and particular kinds of textured patterns.

Appendix - Estimating phase when queries contain multiple images

When matching multiple images in a query to multiple images in a database the measured phase depends on the encoded phases of the query and database images. The phase at the peak location derived from the $\mathbf{z}_+$ correlation surface

$$\phi_+ = \phi_k + \phi_m \qquad \text{(A-1)}$$

where ϕ_k and ϕ_m are the encoded phases of the k-th database image and the m-th query image. Correlating the phase encoded data with the conjugate phase encoded query produces the $\mathbf{z}_-$ correlation surface where the phase at the peak location is

$$\phi_- = \phi_k - \phi_m . \qquad \text{(A-2)}$$

As shown in Fig. A-1, ϕ_k is equal to the average phase angle ±180¡

$$\phi_k = \begin{cases} (\phi_+ + \phi_-)/2 \\ \pi + (\phi_+ + \phi_-)/2 \end{cases} \tag{A-3}$$

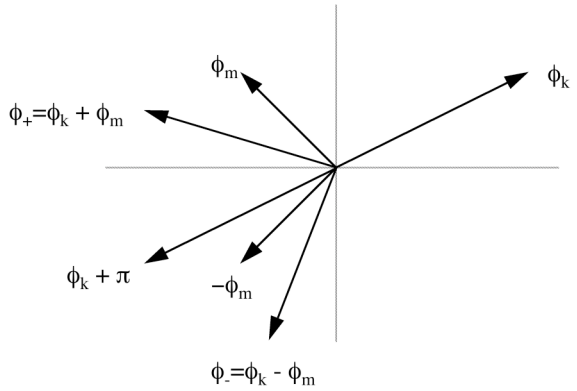

Fig. A-1 Geometry for phase estimation

By subtracting the measured phases and dividing by two,

$$\phi_m = \begin{cases} (\phi_+ - \phi_-)/2 \\ \pi + (\phi_+ - \phi_-)/2 \end{cases} \tag{A-4}$$

we can determine which image in the query (modulo M) matched the database.

References

1. M. Carlotto, "Alternative access methods for image retrieval," Earth and Space Science Information Systems, AIP Conference Proceedings, Vol. 238, pp 527-538, Feb. 1992, American Institute of Physics.

2. W. Niblack et al, "The QBIC project: Querying images by content using color, texture, and shape," SPIE Proc. Storage and Retrieval of Image and Video Databases, Vol. 1908, 1993.

3. SPIE Proc. Storage and Retrieval of Image and Video Databases, Vol. 1908, 1993.

4. SPIE Proc. Storage and Retrieval of Image and Video Databases II, Vol. 2185, 1994.

5. SPIE Proc. Storage and Retrieval of Image and Video Databases III, Vol. 2420, 1995.

6. L. Hassebrook, B.V.K. Vijaya Kumar and L. Hostetler, "Linear phase coefficient composite filters for optical pattern recognition," SPIE Proc. Optical Pattern Recognition, Vol. 1053, pp 218-226 1989.

7. L. Hassebrook, M. Rahmati and B.V.K. Vijaya Kumar, "Hybrid composite filter bands for distortion-invariant optical pattern recognition," Optical Engineering, Vol. 31, No. 5, pp 923-933, May 1992.

8. A.V. Oppenheim and J.S. Lim, "The importance of phase in signals," Proc. IEEE, Vol. 69, No. 5, May 1981.