A Fast and Exact Modulo-Distance Between Histograms

Francesc Serratosa[1] and Alberto Sanfeliu[2]

[1] Universitat Rovira I Virgili, Dept. d'Enginyeria Informàtica i Matemàtiques, Spain
francesc.serratosa@.urv.net
[2] Universitat Politècnica de Catalunya, Institut de Robòtica i Informàtica Industrial, Spain
sanfeliu@iri.upc.es

Abstract. The aim of this paper is to present a new method to compare modulo histograms. In these histograms, the type of elements are cyclic, for instance, the hue in colour images. The main advantage is that there is an important time-complexity reduction respect the methods presented before. The distance between histograms that we present is defined on a structure called *signature*, which is a lossless representation of histograms.

We show that the computational cost of our distance is $O(z'^2)$, being z' the number of non-empty bins of the histograms. The computational cost of the algorithms presented in the literature depends on the number of bins of the histograms. In most of the applications, the obtained histograms are sparse, then considering only the non-empty bins makes the time consuming of the comparison drastically decrease.

The distance and algorithms presented in this paper are experimentally validated on the comparison of images obtained from public databases.

1 Introduction

A histogram of a set with respect to a measurement represents the frequency of quantified values of that measurement among the samples. Finding the distance or similarity between histograms is an important issue in pattern classification or clustering and image retrieval. For this reason, a number of measures of similarity between histograms have been proposed and used in computer vision and pattern recognition. Protein classification is one of the common histogram applications [9]. Moreover, if the ordering of the elements in the sample is unimportant, the histogram obtained from this set is a lossless representation of it and can be reconstructed from its histogram. Then, we can compute the distance between sets in an efficient way by computing the distance between their histograms.

The probabilistic approaches use histograms based on the fact that the histogram of a measurement provides the basis for an empirical estimate of the probability density function [1]. Computing the distance between probability density functions can be regarded as the same as computing the Bayes probability. This is equivalent to measuring the overlap between probability density functions as the distance. The *B-distance* [2], proposed by Kailath, measures the distance between populations. It is a value between 0 and 1 and provides bounds on the Bayes misclassification probability. An approach closely related to the *B-distance* was proposed by Matusita [3]. Finally, Kullback generalised the concept of probabilistic uncertainty or

D.-Y. Yeung et al. (Eds.): SSPR&SPR 2006, LNCS 4109, pp. 394 – 402, 2006.

"entropy" and introduced the *K-L-distance* measure [1,4] that is the minimum cross entropy.

Most of the distance measures presented in the literature (there is an interesting compilation in [5]) consider the overlap or intersection between two histograms as a function of the distance value but they do not take into account the similarity on the non-overlapping parts of the two histograms. For this reason, Rubner presented in [6] a new definition of the distance measure between histograms that overcomes this non-overlapping parts problem. It was called Earth Mover's Distance and it is defined as the minimum amount of work that must be performed to transform one histogram into the other one by moving distribution mass. They used the simplex algorithm [8] to compute the distance measure and the method presented in [7] to search a good initialisation.

We consider three types of measurements called nominal, ordinal and modulo. In a nominal measurement, each value of the measurement is a name an there is not any relation between them such as great than or lower than (e.g. the names of the students). In an ordinal measurement, the values are ordered (e.g. the age of the students). Finally, in the modulo measurement, measurement values are ordered but form a ring due to the arithmetic modulo operation (e.g. the angle in a circumference).

Cha presented in [5] three new algorithms to obtain the distance between one-dimensional histograms that use the Earth Mover's Distance. These algorithms computed the distance between histograms when the type of measurements where *nominal*, *ordinal* and *modulo* in $O(z)$, $O(z)$ and $O(z^2)$ respectively, being z the number of levels or bins.

Often, for specific set measurements, only a small fraction of the *bins* in a histogram contain significant information, that is, most of the *bins* are empty. This is more frequent when the dimensions of the element domain increase. In that cases, the methods that use histograms as fixed-sized structures obtain poor efficiency. For this reason, Rubner [6] presented the variable-size descriptions called *signatures*. In that representations, the empty bins where not explicitly considered.

Another method used to reduce the dimensionality of the data in the case that the statistical properties of the data are a priori known was shown in [10]. The similarity measures are improved by the smoothing projections that are applicable for reduction of the dimensionality of the data and also to represent sparse data in a more tight form in the projection subspace.

We presented in [12] the definition of the nominal, ordinal and modulo distances between histograms in which, only the non-empty bins where considered. In [11], the algorithms of these distances where shown, demonstrated and validated.

In this paper, we present the algorithm to compute the modulo distance between histograms that the computational cost depends only on the non-empty bins instead of the number of bins as it is in the algorithms presented in [5,6]. The time saving of our modulo-distance algorithm is higher than our nominal-distance or ordinal-distance algorithms due to the computational cost is quadratic instead of lineal.

The subsequent sections are constructed as follows. First, we define the histograms and signatures. Then in section 3 we define the modulo distance between signatures. In section 4, we depict the basic algorithm to compute the modulo distance between signatures. In section 5, we use our method to compare images obtained from databases. Finally, we conclude with emphasis of the advantage of using our distance between signatures and using the proposed algorithm.

2 Histograms and Signatures

In this section, we formally give a definition of histograms and signatures. The section finishes with a simple example to show the representations of the histograms and signatures given a set of measurements.

2.1 Histogram Definition

Let x be a measurement which can have one of T values contained in the set $X=\{x_1,...x_T\}$. Consider a set of n elements whose measurements of the value of x are $A=\{a_1,...a_n\}$ where $a_t \in X$.

The histogram of the set A along measurement x is $H(x,A)$ which is an ordered list consisting of the number of occurrences of the discrete values of x among the a_t. As we are interested only in comparing the histograms and sets of the same measurement x, $H(A)$ will be used instead of $H(x,A)$ without loss of generality. If $H_i(A)$, $1 \le i \le T$, denotes the number of elements of A that have value x_i, then $H(A)=[H_1(A), ...,H_T(A)]$ where

$$H_i(A) = \sum_{t=1}^{n} C_{i,t}^{A} \tag{1}$$

and the individual costs are defined as

$$C_{i,t}^{A} = \begin{cases} 1 & if\ a_t = x_i \\ 0 & otherwise \end{cases} \tag{2}$$

The elements $H_i(A)$ are usually called *bins* of the histogram.

2.2 Signature Definition

Let $H(A)=[H_1(A), ...,H_T(A)]$ and $S(A)=[S_1(A), ...,S_z(A)]$ be the histogram and the signature of the set A, respectively. Each $S_k(A)$, $1 \le k \le z \le T$ is composed by a pair of terms, $S_k(A)=\{w_k, m_k\}$. The first term, w_k, shows the relation between the signature $S(A)$ and the histogram $H(A)$. Thus, if the $w_k=i$ then the second term, m_k, is the number of elements of A that have value x_i, that is, $m_k=H_i(A)$ where $w_k<w_t \Leftrightarrow k<t$ and $m_k>0$.

The signature of a set is a lossless representation of its histogram in which the *bins* of the histogram that has value 0 are not expressed implicitly. From the signature definition, we obtain the following expression,

$$H_{w_k}(A) = m_k \quad where \quad 1 \le k \le z \tag{3}$$

2.3 Extended Signature

The **extended signature** is a signature in which the minimum number of empty bins has been added to assure that, given a pair of signatures to be compared, the number of bins is the same. Moreover, each bin in both signatures represents the same bin in the histograms.

2.4 Example

In this section we show a pair of sets with their histogram and signature representations. This example is used to explain the distance measures in the next sections. Figure 1 shows the sets *A* and *B* and their histogram representations. Both sets have 10 elements and values are contained from 1 to 8. Horizontal axis in the histograms represents the values of the elements and the vertical axis represents the number of elements that have this value, that is m_i. Empty bins are the ones that $m_i=0$.

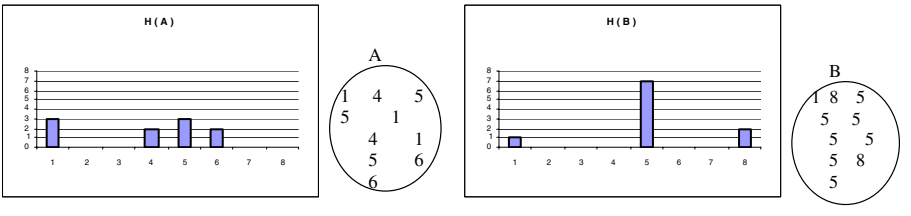

Fig. 1. Sets *A* and *B* and its histograms

Figure 2 shows the signature representation of the sets *A* and *B*. The length of the signatures is 4 and 3, respectively. The vertical axis represents the number of elements of each bin and the horizontal axis represents the bins of the signature. The set A has 2 elements with value 6 since this value is represented by the bin 4 ($W_4^A=6$) and the value of the vertical axis is 2 at bin 4. In the signature representation there is not any empty bin.

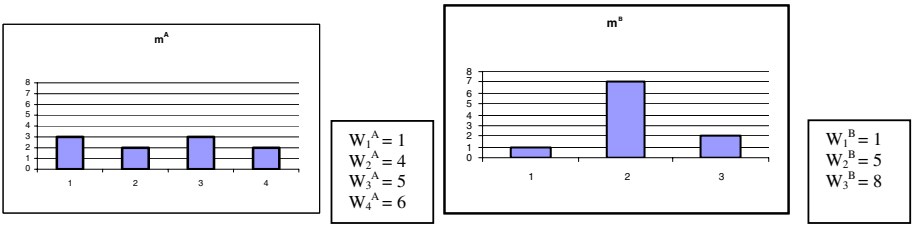

Fig. 2. Signature representation of the sets *A* and *B*

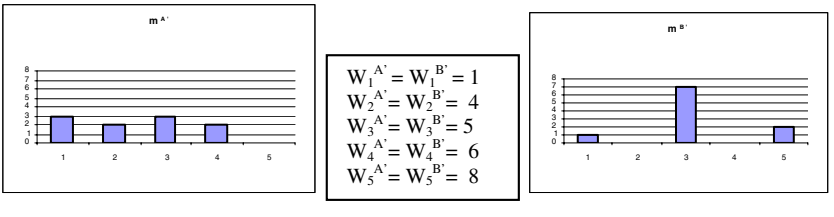

Fig. 3. Extended Signatures *A'* and *B'*. The number of elements m_i are represented graphically and the value of its elements is represented by w_i.

Figure 3 shows the extended signatures of the sets *A* and *B* with *5* bins. Note that the value that the extended signatures represent for each bin, w_i, is the same for both signatures. Moreover, in *A'* and *B'*, one and two empty bins have been added, respectively.

3 Modulo Distance Between Signatures

The aim of this section is to present the new distance between signatures. To do so, we first show de definition of the distance between histograms and then we move on the new definition of the distance between signatures. The algorithms used to obtain the extended signatures and the distances are described in the algorithms section.

For the following definition of the distance and also for the algorithms section, we assume that the extended signatures of *S(A)* and *S(B)* are *S(A')* and *S(B')*, respectively, where $S_i(A') = \{w_i^{A'}, m_i^{A'}\}$ and $S_i(B') = \{w_i^{B'}, m_i^{B'}\}$. The number of bins of *S(A)* and *S(B)* is z^A and z^B and the number of bins of both extended signatures is *z'*.

The distance value between two modulo measurement values is the interior difference of each element (see [5] for the proofs the metric property).

$$d_{\text{mod}}(a,b) = \begin{cases} |a-b| & \text{if } |a-b| \le T/2 \\ T-|a-b| & \text{otherwise} \end{cases} \tag{4}$$

The modulo distance between two histograms was presented in [6] as the minimum of work needed to transform one histogram to the other. Histogram *H(A)* can be transformed into histogram *H(B)* by moving elements to left or right and the total of all necessary minimum movements is the distance between them. There are two operations. Suppose an element *a* that belong to the bin *i*. One operation is *move left (a)*. This operation results that the element *a* belong to bin *i-1* and the cost to do so is 1. Another operation is *move right (a)*. Similarly, after the operation, *a* belongs to the bin *i+1* and the cost is 1. These operations are graphically represented by right-to-left arrows and left-to-right arrows.

In a modulo type histograms, the first bin and the last bin are considered to be adjacent to each other, and hence, it forms a closed circle, due to the nature of the data type. Transforming a modulo type histogram to another while computing their distance

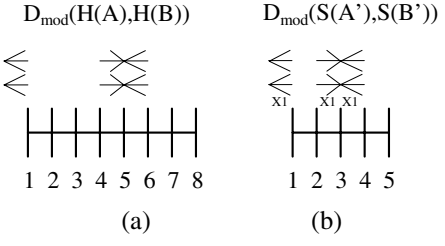

Fig. 4. Arrow representation of the modulo distance using (a) histograms and (b) signatures

should allow cells to move from the first bin to the last one or vice versa at a cost of a single movement. Thus, cells or blocks of earth can move from the fist bin to the last bin with the operation *move left (1)*. Similarly, blocks can move from the last bin to the first one with the operations *move right (T)*.

Figure 4 shows the arrows needed to transform (a) histogram *H(A)* to histogram *H(B)* and (b) the extended signature *S(A')* to *S(B')*.

The main difference between the histogram and signature case is that in the second one we have to take into consideration that the difference between bins is not constant. Our arrows have not a constant size (or constant cost) but they depend on the distance between bins. If element *a* belongs to the bin *i*, the operation *move left (a)* results that the element *a* belong to bin *i-1* and the cost to do so is $w_i - w_{i-1}$. Similarly, after the operation *move right(a)*, the element *a* belongs to the bin *i+1* and the cost is $w_{i+1} - w_i$.

The costs of the last and first movements are the addition of three terms. (a) The cost from the last bin of the signature, $w_{z'}$, to the last bin of the histogram, *T*. (b) The cost from the last bin of the histogram, *T*, to the first bin of the histogram, *1*. (c) The cost from the first bin of the histogram, *1*, to the first bin of the signature, w_1. Then, the costs are calculated as the length of these terms. The cost of (a) is $T-w_{z'}$, the cost of (b) is *1* (similarly to the cost between histograms) and the cost of (c) is w_1-1. Therefore, the final cost from the last bin to the first one or vice versa between signatures is $w_1-w_{z'}+T$.

Due to the modulo properties explained before, we can transform one signature or histogram into another one in several ways. Among these ways, there exists a minimum distance whose number of movements (or the cost of the arrows and the number of arrows) is the lowest. If there is a border line between bins that has both directional arrows, they are cancelled out. These movements are redundant and so the distance cannot be obtained through this configuration of arrows. To find the minimum configuration of arrows, we can add a complete chain in the histogram or signature of same directional arrows, then the opposite arrows on the same border between bins are cancelled out.

The modulo distance between signatures is defined as follows,

$$D_{\mathrm{mod}}\left(S(A),S(B)\right)=\min_{c}\left\{\sum_{i=1}^{z'-1}\left[\left(w_{i+1}^{A'}-w_i^{A'}\right)\left|c+\sum_{j=1}^{i}\left(m_j^{A'}-m_j^{B'}\right)\right|\right]+\left(w_1^{A'}-w_{z'}^{A'}+T\right)|c|\right\} \quad (5)$$

The cost of the movement of blocks from the first bin to the last one or viceversa is $w_1-w_{z'}+T$ and the costs of the other movements is $w^{A'}{}_{i+1}-w^{A'}{}_i$. Moreover, *c* represents the chains of left arrows or right arrows added to the current arrow representation. The absolute value of *c* at the end of the expression is the number of chains added to the current representation. It is multiplied by the cost of the arrows from the last bin to the first one or vice versa.

Example. Figure 5 shows five different transformations of signature *S(A)* to signature *S(B)* and their related costs. The cost is the number of arrows multiplied by the length of the arrows (shown under the arrows). In the first transformation, one chain of right

arrows are added (c=1). In the second one, no chains are added (c=0), thus the cost is the same than the ordinal distance. In the third to the last ones, 1, 2 and 3 chains of left arrows are added, respectively. We can see that the minimum cost is 6 and it is the case that c=-2, then the distance value is 6 for the modulo distance and 14 for the ordinal distance.

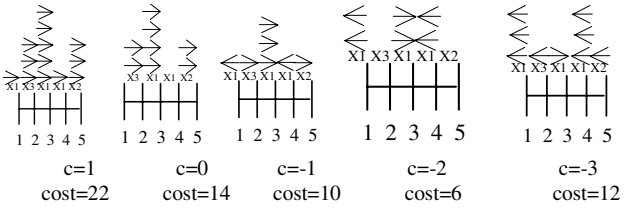

Fig. 5. Five different transformations of signature *S(A)* to the signature *S(B)* with their related *c* and the obtained cost

4 Algorithm

The process *Modulo_Distance* obtains the modulo distance of two signatures. Given two signatures, the process *Extended_Signature* obtains two minimum extended signatures in *O(z)* (the algorithm was presented in [11]).

```
Dmod = Modulo_Distance {S(A),S(B)}
{S(A'),S(B'),z'} = Extended_Signature {S(A),S(B)}
1.  Dmod = 0 p[0] = m0^A' - m0^B'
2.  for (i = 2 to z') p[i] = mi^A' - mi^B' + p[i-1]
3.  for (i = 1 to z-1') Dmod += (wi+1^A' - wi^A') * abs(p[i])
4.  do
5.     D2=0
6.     c = min positive {p[i] for 1≤i≤z'}
7.     Temp[i]=p[i]-c for 1≤i≤z'
8.     for (i = 1 to z'-1) D2 += (wi+1^A' - wi^A') * abs(Temp[i])
9.     if (Dmod > D2) Dmod = D2
10.    p[i]= Temp [i] for 1≤i≤z'
11. while(Dmod > D2)
12. do
13.    D2=0
14.    c = max negative {p[i] for 1≤i≤z'}
15.    Temp[i]=p[i]-c for 1≤i≤z'
16.    for (i = 1 to z'-1) D2 += (wi+1^A' - wi^A') * abs(Temp[i])
17.    if (Dmod > D2) Dmod = D2
18.    p[i]= Temp [i] for 1≤i≤z'
19. while(Dmod > D2)
```

Correctness of the Procedure

The arrow representation of minimum distance can be achieved from any arbitrary valid arrow representation by combination of two basic operations: Increasing the chains of right arrows (when the value of *c* is positive) or increasing the chains of left arrows (when the value of *c* is negative). The distance value can increase infinitely

but there exists only one minimum among valid representations. In order to reach to the minima, first the algorithms tests for increasing positively c if whether it gives higher or lower distance value. If the distance reduces, keep applying the operations until no more reduction occurs. Then, the algorithms does the same operations but increasing negatively c. With these two actions, the algorithm guarantees that all possible combinations of correct representations of arrows are tested.

The procedure runs in $O(z'^2)$ time. The lines 1 to 3 obtain the ordinal distance. In the lines 4 – 11 chains of right arrows are added to the current arrow representation until there is no more reduction to the total number of arrows. This increment is considered in the algorithm by the variable c. Next, chains of left arrows are added in the similar manner (lines 12 – 19).

5 Validation of the Method and Algorithm

The method and algorithms presented in this paper are applied on histograms, independently on the kind of the original set from which they have been obtained, i.e. images [13], discretized probability-density functions [14],… The only condition to use our method is to know the type of elements of the original set: ordinal, nominal or modulo.

Table 1. Hue 2^8 bins. Modulo histogram

	Length	Increase Speed	Correct.	Decrease Correct.
Histo.	265	1	86%	1
Signa.	215	1.23	86%	1
Signa. 100	131	2.02	85%	0.98
Signa. 200	95	2.78	73%	0.84
Signa. 300	45	5.88	65%	0.75

Table 2. Hue 2^{16} bins. Modulo histogram

	Length	Increase Speed	Correct.	Decrease Correct.
Histo.	65,536	1	89%	1
Signa.	205	319.68	89%	1
Signa. 1	127	516.03	89%	1
Signa. 2	99	661.97	78%	0.87
Signa. 3	51	1285.01	69%	0.77

To show the validity of our new method, we have tested the modulo distance between histograms and between signatures. We used 1000 images (640 x 480 pixels) obtained from public databases. To validate the modulo distance, the histograms represent the hue coordinate with 2^8 levels (table 1) and with 2^{16} levels (table 2). Each of the tables below shows the results of 5 different tests. In the first and second files of the tables, the distance where computed between histograms and signatures, respectively. In the other three, the distance was computed between signatures but, with the aim of reducing the length of the signature (and so to increase the speed), the bins that had less elements than 100, 200 or 300 in table 1 and less elements than 1, 2 or 3 in table 2 where removed. The first column is the number of bins of the histogram (first cell) or signatures (the other four cells). The second column represents the increase of speed if we use signatures respect histograms. It is calculated as the ratio between the run time of the histogram method and the signature method. The third column is the average correctness. The last column represents the decrease of correctness due to using the signatures with filtered histograms. It is obtained as the ratio of the correctness of the histogram by the correctness of each filter.

Tables 1 and 2 show us that our method is much useful when the number of bins increases since the number of empty bins tends to increase. Note that in the case of the first filter (third experiment in the tables), there is no decrease in the correctness although there is much increase in the speed respect the signature method (second experiment in the tables).

6 Conclusions and Future Work

We have presented the modulo distance between signatures and the algorithm used to compute it. We have shown that signatures are a lossless representation of histograms and that computing the distance between signatures is the same than between histograms but with a lower computational time. We have validated this new algorithm with a huge amount of real images and we have realised that there is an important time saving do to most of the histograms are sparse. Moreover, if we apply filtering techniques on the histograms, the number of bins of the signatures reduces and so the run time of their comparison.

References

1. R.O. Duda, P.E. Hart & D.G. Stork, *Pattern Classification*, 2nd edition, Wiley, New York, 2000.
2. T. Kailath, "The divergence and bhattacharyya distance measures in signal selection", *IEEE Transactions Community Technol*. COM-15, 1, pp:52-60, 1967.
3. K. Matusita, "Decision rules, based on the distance, for problems of fit, two samples and estimation", *Annals Mathematic Statistics*, 26, pp: 631-640, 1955.
4. J.E. Shore & R.M. Gray, "Minimum cross-entropy pattern classification and cluster analysis", Transactions on Pattern Analysis and Machine Intelligence, 4 (1), pp: 11-17, 1982.
5. S.-H. Cha, S. N. Srihari, "On measuring the distance between histograms" *Pattern Recognition* 35, pp: 1355–1370, 2002.
6. Y. Rubner, C. Tomasi, and L. J. Guibas, "A Metric for Distributions with Applications to Image Databases" *International Journal of Computer Vision* 40 (2), pp: 99-121, 2000.
7. E. J. Russell. "Extension of Dantzig's algorithm to finding an initial near-optimal basis for the transportation problem", *Operations Research*,17,pp:187-191, 1969.
8. *Numerical Recipes in C: The Art of Scientific Computing*, ISBN 0-521-43108-5.
9. Y-P Nieh & K.Y.J. Zhang, "A two-dimensional histogram-matchinig method for protein phase refinement and extension", Biological Crystallography, 55, pp:1893-1900, 1999.
10. J.-K. Kamarainen, V. Kyrki, J. Llonen, H. Kälviäinen, "Improving similarity measures of histograms using smoothing projections", *Pattern Recognition Letters* 24, pp: 2009–2019, 2003.
11. F. Serratosa & A. Sanfeliu, "Signatures versus Histograms: Definitions, Distances and Algorithms", *Pattern Recognition* (39), Issue 5, pp. 921-934, 2006.
12. F. Serratosa & A. Sanfeliu, "A fast distance between histograms", *Lecture Notes and Computer Science 3773, pp: 1027 - 1035*, 2005
13. M. Pi, M.K. Mandal, A. Basu, "Image retrieval based on histogram of fractal parameters", *Multimedia, IEEE Transactions on*, Vol. 7 (4), 2005, pp. 597 – 605.
14. F. Serratosa, R. Alquézar y A. Sanfeliu, "Function-Described Graphs for modeling objects represented by attributed graphs", *Pattern Recognition*, 36 (3), pp. 781-798, 2003.