HIDDEN MARKOV MODELS FOR TWO-DIMENSIONAL DATA

Janusz Bobulski[1]

Abstract Hidden Markov models are well-known methods for image processing. They are used in many areas where 1D data are processed. In the case of 2D data, there appear some problems with application HMM. There are some solutions, but they convert input observation from 2D to 1D, or create parallel pseudo 2D HMM, which is set of 1D HMMs in fact. This paper describes authentic 2D HMM with two-dimensional input data, and its application for pattern recognition in image processing.

1 Introduction

Hidden Markov models (HMM) are widely apply in data classification. They are used in speech recognition, character recognition, biological sequence analysis, financial data processing, texture analysis, face recognition, etc. This widely application of HMM is result of its effectiveness. An extension of the HMM to work on two-dimensional data is 2D HMM. A 2D HMM can be regarded as a combination of one state matrix and one observation matrix, where transition between states take place according to a 2D Markovian probability and each observation is generated independently by the corresponding state at the same matrix position. It was noted that the complexity of estimating the parameters of a 2D HMMs or using them to perform maximum a posteriori classification is exponential in the size of data. Similar to 1D HMM, the most important thing for 2D HMMs is also to solve the three basic problems, namely, probability evolution, optimal state matrix and parameters estimation.

Czestochowa University of Technology
Institute of Computer and Information Science
Dabrowskiego Street 73, 42-200 Czestochowa, Poland `januszb@icis.pcz.pl`

When we process one-dimensional data, we have good tools and solution for this. Unfortunately, this is unpractical in image processing, because the images are two-dimensional. When you convert an image from 2D to 1D , you lose some information. So, if we process two-dimensional data, we should apply two-dimensional HMM, and this 2D HMM should work with 2D data. One of solutions is pseudo 2D HMM [2, 6, 10]. This model is extension of classic 1D HMM. There are super-states, which mask one-dimensional hidden Markov models (Fig. 1). Linear model is the topology of superstates, where only self transition and transition to the following superstate are possible. Inside the superstates there are linear 1D HMM. The state sequences in the rows are independent of the state sequences of neighboring rows. Additional, input data are divided to the vector. So, we have 1D model with 1D data in practise.

Other approach to image processing use two-dimensional data present in

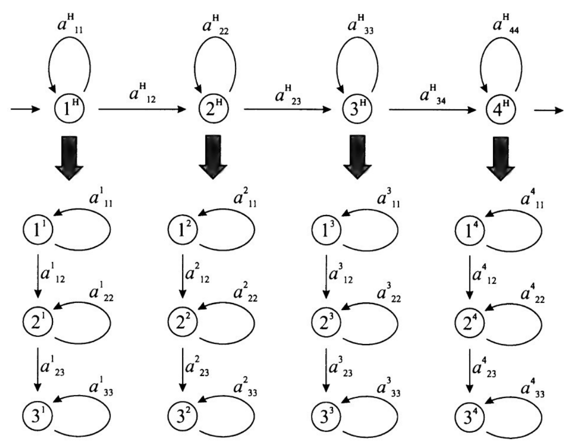

Fig. 1 Pseudo 2D HMM [1].

works [4] and [7]. The solutions base on Markov Random Fields (MRF) and give good results for classification and segmentation, but not in pattern recognition. Interesting results showed in paper [11]. This article presents analytic solution and evidence of correctness two-dimensional HMM. But this 2D HMM is similar to MRF, works with one-dimensional data and can be apply only for left-right type of HMM. This article presents real solution for 2D problem in HMM. There is show true 2D HMM which processes 2D data.

2 Classic 1D HMM

HMM is a double stochastic process with underlying stochastic process that is not observable (hidden), but can be observed through another set of stochastic processes that produce a sequence of observation [8]. Let $O = \{O_1, .., O_T\}$ be the sequence of observation of feature vectors, where T is the total number of feature vectors in the sequence. The statistical parameters of the model may be defined as follows [5, 9]:

- The number of states of the model, N
- The number of symbols M
- The transition probabilities of the underlying Markov chain, $A = \{a_{ij}\}, 1 \leq i, j \leq N$, where a_{ij} is the probability of transition from state i to state j
- The observation probabilities, $B = \{b_{jm})\} \quad 1 \leq j \leq N, 1 \leq m \leq M$ which represents the probability of gnerate the m_{th} symbol in the j_{th} state.
- The initial probability vector, $\Pi = \{\pi_i\} \quad 1 \leq i \leq N$.

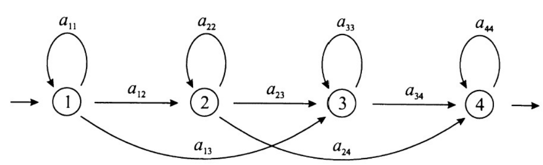

Fig. 2 One-dimensional HMM.

Hence, the HMM requires three probability measures to be defined, A, B, Π and the notation $\lambda = (A, B, \Pi)$ is often used to indicate the set of parameters of the model. In the proposed method, one model is made for each picture of pattern. The parameters of the model are generated at random at the beginning. Then they are estimated with Baum-Welch algorithm, which is based on the forward-backward algorithm. The forward algorithm calculates the coefficient $\alpha_t(i)$ (probability of observing the partial sequence $(o_1, , o_t)$ such that state q_t is i). The backward algorithm calculates the coefficient $\beta_t(i)$ (probability of observing the partial sequence $(o_{t+1}, , o_T)$ such that state q_t is i). The Baum-Welch algorithm, which computes the λ, can be described as follows [5]:

1. Let initial model be λ_0
2. Compute new λ based on λ_0 and observation O
3. If $log(P(O|\lambda) - log(P(O)|\lambda_0) < DELTA$ stop
4. Else set $\lambda \rightarrow \lambda_0$ and go to step 2.

The parameters of new model λ, based on λ_0 and observation O, are estimated from equation of Baum-Welch algorithm [9], and then are recorded to the database.

3 Three basic problems

There are three fundamental problems of interest that must be solved for HMM to be useful in some applications. These problems are the following:

1. Given observation $O = (o_1, o_2, , o_T)$ and model $\lambda = (A, B, \Pi)$, efficiently compute $P(O|\lambda)$
2. Given observation $O = (o_1, o_2, , o_T)$ and model λ find the optimal state sequence $q = (q_1, q_2, , q_T)$
3. Given observation $O = (o_1, o_2, , o_T)$, estimate model parameters $\lambda = (A, B, \Pi)$ that maximize $P(O|\lambda)$

3.1 Solution to Problem 1

Forward Algorithm [5]:

- Define forward variable $\alpha_t(i)$ as:

$$\alpha_t(i) = P(o_1, o_2, , o_t, q_t = i|\lambda) \tag{1}$$

- $\alpha_t(i)$ is the probability of observing the partial sequence $(o_1, o_2, , o_t)$ such that the the state q_t is i
- Induction
  1. Initialization:
  $$\alpha_1(i) = \pi_i b_i(o_1) \tag{2}$$
  2. Induction:
  $$\alpha_{t+1}(i) = \left[\sum_{i=1}^{N} \alpha_t(i) a_{ij}\right] b_j(o_{t+1}) \tag{3}$$
  3. Termination:
  $$P(O|\lambda) = \sum_{i=1}^{N} \alpha_T(i) \tag{4}$$

Backward Algorithm [5]:

- Define backward variable $\beta_t(i)$ as:

$$\beta_t(i) = P(o_{t+1}, o_{t+2}, , o_T, q_t = i|\lambda) \tag{5}$$

- $\beta_t(i)$ is the probability of observing the partial sequence $(o_1, o_2, , o_t)$ such that the the state q_t is i
- Induction
  1. Initialization:
  $$\beta_T(i) = 1 \tag{6}$$

2. Induction:

$$\beta_t(i) = \sum_{i=1}^{N} a_{ij} b_j(o_{t+1}\beta_{t+1}(j), \tag{7}$$

$$1 \leq i \leq N, t = T-1, ..., 1$$

3. Termination:

$$P(O|\lambda) = \sum_{i=1}^{N} \beta_1(i) \tag{8}$$

3.2 Solution to Problem 2

Viterbi Algorithm [5]:

- Initialization:

$$\delta_1(i) = \pi_i b_i(o_1), 1 \leq i \leq N \tag{9}$$

$$1 \leq i \leq N$$

$$\psi_1 = 0 \tag{10}$$

- Recursion:

$$\delta_t(j) = \max_{1 \leq i \leq N} [\delta_{t-1}(i) a_{ij}] b_j(o_t) \tag{11}$$

$$\psi_t(j) = arg \max_{1 \leq i \leq N} [\delta_{t-1}(i) a_{ij}] b_j(o_t) \tag{12}$$

$$1 \leq j \leq N, 2 \leq t \leq T$$

- Termination:

$$P^* = \max_{1 \leq i \leq N} [\delta_t(i)] \tag{13}$$

$$q_t^* = arg \max_{1 \leq i \leq N} [\delta_t(i)] \tag{14}$$

- Backtracking:

$$q_t^* = \psi_t(q_{t+1}^*) \tag{15}$$

$$t = T-1, T-2, ..., 1$$

3.3 Solution to Problem 3

Baum-Welch Algorithm [5]:

- Define $\xi(i,j)$ as the probability of being in state i at time t and in state j at time $t+1$

$$\xi(i,j) = \frac{\alpha_t(i)a_{ij}b_j(o_{t+1})\beta_{t+1}(j)}{P(O|\lambda)} = \frac{\alpha_t(i)a_{ij}b_j(o_{t+1})\beta_{t+1}(j)}{\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_t(i)a_{ij}b_j(o_{t+1})\beta_{t+1}(j)} \tag{16}$$

- Define $\gamma(i)$ as the probability of being in state i at time t, given observation sequence.

$$\gamma_t(i) = \sum_{j=1}^{N} \xi_t(i,j) \tag{17}$$

- $\sum_{t=1}^{T}\gamma_t(i)$ is the expected number of times state i is visited
- $\sum_{t=1}^{T-1}\xi_t(i,j)$ is the expected number of transition from state i to j

Update rules:

- $\bar{\pi}_i$ = expected frequency in state i at time $(t=1) = \gamma_1(i)$
- $\bar{a}_{ij}$ = (expected number of transition from state i to state j)/(expected number of transitions from state i:

$$\bar{a}_{ij} = \frac{\sum_t \xi_t(i,j)}{\sum_t \gamma_t(i)} \tag{18}$$

- $\bar{b_j}(k)$ = (expected number of times in state j and oserving symbol k)/(expected number of times in state j:

$$\bar{b}_j(k) = \frac{\sum_{t,o_t=k}\gamma_t(j)}{\sum_t \gamma_t(j)} \tag{19}$$

4 2D HMM

In paper [11], Yujian proposed definitions and proofs of 2D HMM. He has presented several analytic formulae for solving the three basic problems of 2-D HMM. Solution to Problem 2 is usefull, and Viterbi algorithm can be easily adopted to image recognition with two dimensional input data. Unfortunetly, solution to problem 1 and 3 may be use only with one dimensional data - observation vector. Besides presented solutions are for Markov model type "left-right", and not ergodic. So, I present solution to problems 1 and 3 for two dimensional data. The statistical parameters of the 2D model (Fig. 3):

- The number of states of the model N^2
- The number of data streams k_1 x $k_2 = K$
- The number of symbols M
- The transition probabilities of the underlying Markov chain, $A = \{a_{ijl}\}, 1 \leq i,j \leq N, 1 \leq l \leq N^2$, where a_{ij} is the probability of transition from state ij to state l

- The observation probabilities, $B = \{b_{ijm})\}, 1 \leq i, j \leq N, 1 \leq m \leq M$ which represents the probability of gnerate the m_{th} symbol in the ij_{th} state.
- The initial probability, $\Pi = \{\pi_{ijk}\}, 1 \leq i, j \leq N, 1 \leq k \leq K$.
- Oservation sequance $O = \{o_t\}, 1 \leq t \leq T, o_t$ is square matrix simply observation with size k_1 x $k_2 = K$

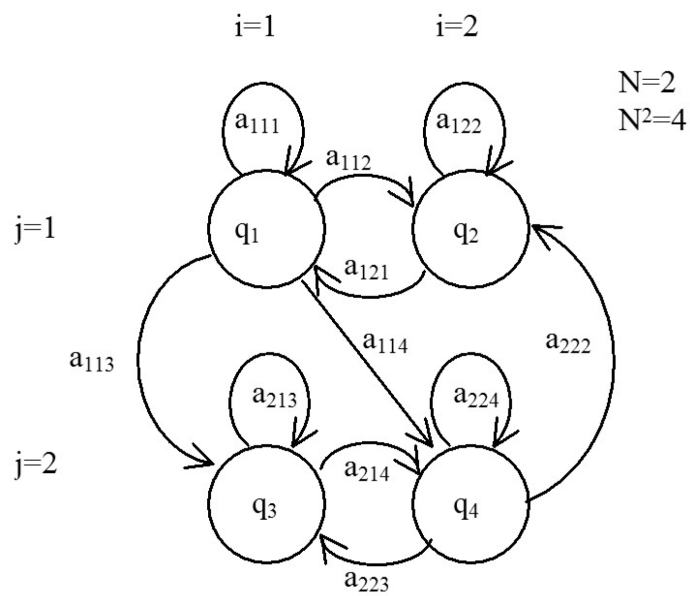

Fig. 3 Two-dimensional ergodic HMM.

4.1 Solution to 2D Problem 1

Forward Algorithm

- Define forward variable $\alpha_t(i, j, k)$ as:

$$\alpha_t(i, j, k) = P(o_1, o_2, , o_t, q_t = ij|\lambda) \tag{20}$$

- $\alpha_t(i, j, k)$ is the probability of observing the partial sequence $(o_1, o_2, , o_t)$ such that the the state q_t is i, j for each k_{th} strem of data
- Induction

1. Initialization:
$$\alpha_1(i,j,k) = \pi_{ijk} b_{ij}(o_1) \tag{21}$$
2. Induction:
$$\alpha_{t+1}(i,j,k) = \left[\sum_{l=1}^{N} \alpha_t(i,j,k) a_{ijl}\right] b_{ij}(o_{t+1}) \tag{22}$$
3. Termination:
$$P(O|\lambda) = \sum_{t=1}^{T} \sum_{k=1}^{K} \alpha_T(i,j,k) \tag{23}$$

4.2 Solution to 2D Problem 3

Parameters reestimation Algorithm:

- Define $\xi(i,j,l)$ as the probability of being in state ij at time t and in state l at time $t+1$ for each k_{th} strem of data

$$\xi_t(i,j,l) = \frac{\alpha_t(i,j,k) a_{ijl} b_{ij}(o_{t+1}) \beta_{t+1}(i,j,k)}{P(O|\lambda)} = \frac{\alpha_t(i,j,k) a_{ij} b_{ij}(o_{t+1}) \beta_{t+1}(i,j,k)}{\sum_{k=1}^{K} \sum_{l=1}^{N^2} \alpha_t(i,j,k) a_{ijl} b_{ij}(o_{t+1}) \beta_{t+1}(i,j,k)} \tag{24}$$

- Define $\gamma(i,j)$ as the probability of being in state i,j at time t, given observation sequence.

$$\gamma_t(i,j) = \sum_{l=1}^{N^2} \xi_t(i,j,l) \tag{25}$$

- $\sum_{t=1}^{T} \gamma_t(i,j)$ is the expected number of times state ij is visited
- $\sum_{t=1}^{T-1} \xi_t(i,j,l)$ is the expected number of transition from state ij to l

Update rules:

- $\bar{\pi_i} jk$ = expected frequency in state i,j at time $(t=1) = \gamma_1(i,j)$
- $\bar{a}_{ij}$ = (expected number of transition from state i,j to state l)/(expected number of transitions from state i,j:

$$\bar{a}_{ijl} = \frac{\sum_t \xi_t(i,j,l)}{\sum_t \gamma_t(i,j)} \tag{26}$$

- $\bar{b_{ij}}(k)$ = (expected number of times in state j and oserving symbol k)/(expected number of times in state j:

$$\bar{b}_{ij}(k) = \frac{\sum_{t,o_t=k} \gamma_t(i,j)}{\sum_t \gamma_t(i,j)} \tag{27}$$

5 Experimenting

The image database *Amsterdam Library of Object Images* was used in experimenting. It is a color image collection of one-thousand small objects, recorded for scientific purposes. In order to capture the sensory variation in object recordings, they systematically varied viewing angle, illumination angle, and illumination color for each object, and additionally captured wide-baseline stereo images. They recorded over a hundred images of each object, yielding a total of 110,250 images for the collection [1, 3].
In order to verify the method has been selected fifty objects. Three images for learning and three for testing has been chosen. The 2D HMM has been implemented with parameters $N = 5, N^2 = 25, K = 25, M = 50$. Wavelet transform has been chosen as features extraction technigue. Table 1 presents results of experiments.

Table 1 Comparison of recognition rate

Method	Recognition rate [%]
Eigenvector	86
1D HMM	90
2D HMM	92

6 Conclusion

Article presents a new conception about two-dimensional hidden Markov models. We show solutions of principle problems for ergodic 2D HMM, which may be applied for 2D data. Recognition rate of the method is 92%, which is better than 1D HMM. Furthermore, the advantage of this approach is that there is no need to convert the input two-dimensional image on a one-dimensional data, what simplifies pattern recognition system.

References

1. Amsterdam Library of Object Images, http://www.science.uva.nl/aloi
2. Eickeler S., Mller S., Rigoll G. (1999) High Performance Face Recognition Using Pseudo 2-D Hidden Markov Models, European Control Conference, http://citeseer.ist.psu.edu
3. Geusebroek J.M., Burghouts G.J. and A. W. M. Smeulders (2005) The Amsterdam library of object images, Int. J. Comput. Vision 61(1) 103-112
4. Joshi D., Li J., Wang J.Z. (2006) A computationally Efficient Approach to the estimation of two- and three-dimensional hidden Markov models, IEEE Transactions on Image Processing vol. 15 no 7 1871-1886

5. Kanungo T. (1999) Hidden Markov Model Tutorial, http://www.kanungo.com/software/hmmtut.pdf
6. Kubanek M. (2010) Automatic Methods for Determining the Characteristic Points in Face Image, Lecture Notes in Artificial Intelligence 6114 Part I 523-530
7. Li J., Najmi A., Gray R.M. (2000) Image classification by a two dimensional Hidden Markov model, IEEE Transactions on Signal Processing 48 517-533
8. Rabiner L. R. (1989), A tutorial on hidden Markov models and selected application in speech recognition, Proc. IEEE 77 257-285
9. Samaria F., Young S. (1994) HMM-based Architecture for Face Identification, Image and Vision Computing, Vol. 12 No 8 October 537-583.
10. Vitoantonio Bevilacqua V., Cariello L., Carro G., Daleno D., and Mastronardi G. (2008) A face recognition system based on Pseudo 2D HMM applied to neural network coefficients, Soft Comput. 12 7 February 615-621
11. Yujian L. (2007) An analytic solution for estimating two-dimensional hidden Markov models, Applied Mathematics and Computation 185 810-822