Video Affective Content Recognition Based on Genetic Algorithm Combined HMM

Kai Sun, Junqing Yu

Computer College of Science & Technology, Huazhong University of Science & Technology, Wuhan 430074, China
sunkai@smail.hust.edu.cn, yjqing@hust.edu.cn

Abstract. Video affective content analysis is a fascinating but seldom addressed field in entertainment computing research communities. To recognize affective content in video, a video affective content representation and recognition framework based on Video Affective Tree (VAT) and Hidden Markov Models (HMMs) was proposed. The proposed video affective content recognizer has good potential to recognize the basic emotional events of audience. However, due to Expectation-Maximization (EM) methods like the Baum-Welch algorithm tend to converge to the local optimum which is the closer to the starting values of the optimization procedure, the estimation of the recognizer parameters requires a more careful examination. A Genetic Algorithm combined HMM (GA-HMM) is presented here to address this problem. The idea is to combine a genetic algorithm to explore quickly the whole solution space with a Baum-Welch algorithm to find the exact parameter values of the optimum. The experimental results show that GA-HMM can achieve higher recognition rate with less computation compared with our previous works.

1 Introduction

Intensive research efforts in the field of multimedia content analysis in the past 15 years have resulted in an abundance of theoretical and algorithmic solutions for extracting the content-related information from audiovisual information [1]. However, due to the inscrutable nature of human emotions and seemingly broad affective gap from low-level features, the video affective content analysis is seldom addressed [2]. Several research works have been done to extract affective content from video. One method is to map low-level video features into emotion space. Hanjalic and Xu found in literature connections between some low level features of video data streams and dimensions of the emotions space and made algorithmic models for them [2]. Although this method can be used to locate video affective content or highlights effectively, the recognition of specific affective category is beyond its ability. Another method is using HMMs to recognize video affective content [3]. In this method, empirical study on the relationship between emotional events and low level features was performed and two HMM topologies were created. However, it can’t measure the affective intensity and discriminate fear and anger due to the lack of proper low level features (e.g. audio affective features). Taking these results into account, we proposed a novel video affective content representation and recognition framework based on VAT and HMMs [4]. Firstly, video affective content units in different granularities are located by excitement intensity curve and then mapped to be a Video Affective Tree (VAT). The affective intensity of affective content units in VAT is quantified into several levels from weak to strong (e.g. weak, median and strong) according to the intensity of excitement curve. Video affective content can be conveniently represented by VAT at different granularities. Next, low level affective features that represent emotional characteristics are extracted from these affective content units and observation vectors are constructed subsequently by combining these features. Finally, HMMs-based affective recognizer are trained and tested to recognize the basic emotional events (joy, anger, sadness and fear) using these observation vector sequences.

Our HMMs-based affective recognizer used Baum-Welch algorithm to estimate the model parameters. Expectation-Maximization (EM) methods like the Baum-Welch algorithm are known to converge to an optimum of the solution space, but nothing proves that this optimum is the global one. In fact, these algorithms tend to converge to the local optimum which is the closer to the starting values of the optimization procedure. The traditional remedy to this problem consists in running several times the EM algorithm, using different sets of random starting values, and keeping the best solution. More advanced versions of the EM algorithm (ECM, SEM, ...) can also be used, but they do not constitute a definitive answer to the local optima problem [5]. In this paper, we presented a Genetic Algorithm combined HMM (GA-HMM) to address this problem.

2 Genetic Algorithm Combined HMM (GA-HMM)

A HMM is fully described by three sets of probability distributions: a initial distribution π of the hidden variable X, a transition matrix A describing the relation between successive values of the hidden variable X and a transition matrix B used to describe the relation between successive outputs of the observed variable $O=\{O_1,O_2,...,O_T\}$ (T is the number of the observation vectors). Therefore, the computation of HMM involves three different problems: the estimation of the log-likelihood of the data given the model, the estimation of π, A and B given the data and the estimation of the optimal sequence of hidden states given the model and the data. Due to the structure of the HMM, there is no direct formula to compute the log-likelihood. The problem is solved using an iterative procedure known as the forward-backward algorithm. The estimation of the model parameters is traditionally obtained by an Expectation-Maximization (EM) algorithm known in the speech recognition literature as the Baum-Welch algorithm. Finally, the optimal sequence of hidden states is computed using another iterative procedure called the Viterbi algorithm [6].

GA-HMM is presented here to estimate the model parameters more precisely and efficiently, which is an iterative procedure computing simultaneously several possible solutions (the population). At each iteration, a new population is created by combining the members of the current population using the principles of genetics, i.e., selection, crossover and mutation [7]. This new population has a better probability to come close to the global optimum than the previous population had. This method presents the advantage to allow the exploration of a large part of the solution space, with a very high probability to find the global optimum. On the other hand, once the region of the solution space containing the global optimum has been determined, the robustness of GAs is balanced by the large number of iterations required to reach this optimum.

```
Algorithm 1 Estimation of a HMM.
Random initialization of the population.
Computation of the log-likelihood of each member of the population.
for iterga = 1:IterGA do {Iterations of the genetic algorithm.}
  Computation of the new population.
  Recomputation of the log-likelihood of each member of the population.
  for iterbw = 1:IterBW do {Iterations of the Baum-Welch algorithm.}
    for pop=1:PopSize do {Loop on the population size.}
      Reestimation of $\pi$, A, and B for the pop-th member of the population.
      Recomputation of the log-likelihood of the pop-th member of the population.
    end for
  end for
end for
Computation of the optimal sequence of hidden states (Viterbi algorithm).
```

Fig. 1. Estimation of a HMM.

After each iteration of the genetic algorithm, the members of the population are improved through several iterations of a Baum-Welch algorithm. This method is summarized in Algorithm 1 (Fig. 1). *IterGA* is the number of iterations of the genetic algorithm. *IterBW* is the number of iterations of the Baum-Welch algorithm. *PopSize* is the size of the population used by the genetic algorithm.

```
Algorithm 2 Computation of the new population.
Binary coding of each member of the old (current) population.
for pop = 1:PopSize do
  Selection: Random selection of a member of the old population (based on the
  value of its log-likelihood), to become a member of the new population.
  if pop is even then
    Crossover: Exchange a part of the binary vectors describing the last two
    members included in the new population with probability PCross.
  end if
end for
for pop=1:PopSize do
  Mutation: Change 0s to 1s and vice-versa in the binary vector describing the
  pop-th member of the new population with probability PMut.
end for
if the best member of the old population is not included in the new population then
  Elitism: Replace the first member of the new population by the best member
  of the old population.
end if
Decoding of the binary vectors into real parameters form.
```

Fig. 2. Computation of the new population.

The computation of the new population using the genetic algorithm follows the steps given in Algorithm 2 (Fig. 2). Note that the parameters are recoded in binary form at the beginning of the procedure so that each member of the population is described by a vector of zeros and ones. Members of the old population are selected to be part of the new population according to their log-likelihood. This implies that best members of the old population have a greater probability to be chosen. *PCross* is the probability of crossover (exchange of a part of the parameter values) between two members of the population. *PMut* is the probability of mutation, which is the probability with which a 0 is replaced by a 1, and vice versa, in the binary writing of a member of the new population.

3 GA-HMM Based Video Affective Content Recognizer

We aim to utilize GA-HMM in two distinct ways for video affective content recognition: (*i*) given a sequence of observations extracted from an affective content unit, to determine the likelihood of each model (every basic emotion has its own GA-HMM model) for the input data, (*ii*) given the observation sequences, to train the model parameters. The first problem can be regarded to score how well a given model matches a given observation sequence. The second problem is attempting to optimize the model parameters in order to best describe how a given observation sequence comes about. The observation sequence used to adjust the model parameters is called a training sequence since it is used to "train" the GA-HMM. The training problem is the crucial one for our GA-HMMs based affective recognizer, since it allows us to optimally adapt model parameters to observed training data, i.e., to create best models for real phenomena.

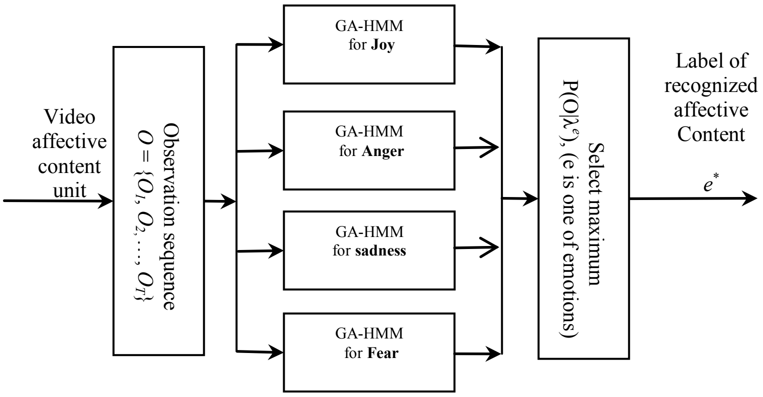
Video affective content unit
Observation sequence $O = \{O_1, O_2, \ldots, O_T\}$
GA-HMM for **Joy**
GA-HMM for **Anger**
GA-HMM for **sadness**
GA-HMM for **Fear**
Select maximum $P(O|\lambda^e)$, (e is one of emotions)
Label of recognized affective Content
e^*

Fig. 3. GA-HMM based affective recognizer.

Therefore, in order to recognize affective content, our affective recognizer contains two components: training and recognizing. For each basic emotion e, we build an ergodic 2-state GA-HMM λ^e. We must estimate the model parameters (A, B, π) that optimize the likelihood of the training set observation vectors for the basic emotion e. For each affective content unit which is to be recognized, the processing listed in

Fig. 3 should be carried out, namely measurement of the observation sequence $O = \{O_1, O_2, ..., O_T\}$ (T is the number of the shots within a affective content unit); followed by calculation of model likelihoods for all of possible models, $P(O \mid \lambda^e)$, $e \in \{joy, anger, sadness, fear\}$; followed by selection of the basic emotion type whose model likelihood is the highest, i.e., $e^* = \underset{e \in \{joy, anger, sadness, fear\}}{\arg\max} \{p(O \mid \lambda^e)\}$. The probability computation step is performed using the Viterbi algorithm (i.e., the maximum likelihood path is used).

4 Experimental Results

We made a data set from ten feature films such as "Titanic", "A Walk in the Clouds", "Love Letter", "Raiders of the Lost Ark", "Con Air", "Pirates of the Caribbean", "Scary Movie", "Spider-Man", "Lord of the Rings" and "the Lion King". Firstly, the excitement intensity curves for all the films were computed. Next, we utilized these excitement intensity curves to construct the Video Affective Tree for each film [4]. The ground truth for the four basic affective events (joy, anger, sadness and fear) was manually determined within the extracted video affective units at different levels. To compute color features, we transformed RGB color space into HSV color space and then quantized the pixels into 11 culture colors such as red, yellow, green, blue, brown, purple, pink, orange, gray, black and white [8]. For a key frame of each shot, we computed the histogram of 11 culture colors. We also computed the saturation(S), value (V), dark colors, and bright colors. So, 15 color features were extracted from each shot. The motion phase and intensity for each shot were also computed. The audio features, such as speech rate, pitch average, pitch range and short time energy, were computed using MPEG-7 XM software [9]. By fusing all of these audio and visual features, observation vectors are generated by vector quantization.

Our experimental results are shown in Table. 1. From the results, it can be easily found that GA-HMM can achieve higher recognition rate compared with classic HMM, which was adopted by our previous works [4].

Table 1. Experimental results

Affective events **Results**		**Joy**	**Anger**	**Sadness**	**Fear**
Number of Affective content units		87	105	152	213
HMM	**Recognized**	76	78	124	153
	Recognition Rate	87.4%	74.3%	81.6%	71.8%
GA-HMM	**Recognized**	81	85	131	167
	Recognition Rate	93.1%	81.0%	86.2%	78.4%

4. Conclusion and Future Work

Hidden Markov model (HMM) is a powerful tool for characterizing the dynamic temporal behavior of video sequences. In this paper, we propose an improved HMM called GA-HMM to design a video affective content recognizer. The experimental results show that GA-HMM can achieve higher recognition rate with less computation compared with our previous works.

Affective content is a subjective concept which relies on audience's perceptions. Talking about affect (feeling, emotion and mood) inevitably calls for a discussion about subjectivity. There are many implementation issues for our proposed video affective content recognizer. For example, finding a representative training data set in the affective domain is a crucial task for us. Moreover, the comparison criteria of models (HMM and GA-HMM) should be further investigated in the real world applications.

Acknowledgments. We gratefully acknowledge the granted financial support from China Postdoctora1 Science Foundation (20060400847) and Huawei Foundation (YJCB20060471N).

References

1. A. Hanjalic: Extracting Moods from Pictures and Sounds: Towards truly personalized TV. IEEE Signal Processing Magazine. 3(2006) 90–100
2. A. Hanjalic, L.-Q. Xu: Affective video content representation and modeling. IEEE Trans. Multimedia. 2(2005) 143–154
3. Hang-Bong Kang: Affective Content Detection using HMMs. Proceedings of the eleventh ACM international conference on Multimedia, pp. 259–262, November 2-8, 2003
4. Kai Sun, Junqing Yu: Video Affective Content Representation and Recognition Using Video Affective Tree and Hidden Markov Models. Lecture Notes in Computer Science. Springer-Verlag, Berlin Heidelberg New York (2007) (to appear)
5. McLachlan, G. J., Krishnan, T.: The EM Algorithm and Extensions. New York: Wiley, 1997.
6. L. Rabiner: A tutorial on hidden Markov models and selected applications in speech recognition. Proc. IEEE, vol. 77, no. 2, pp. 256–286, Feb. 1989
7. Coley, D.A.: An introduction to genetic algorithms for scientists and engineers. World Scientific Press, 1999.
8. Goldstein, E. : Sensation and Perception. Brooks/Cole, 1999
9. Information Technology—Multimedia Content Description Interface—Part 4: Audio, ISO/IEC CD 15938-4, 2001.