A MATHEMATICAL APPROACH TOWARDS SEMI-AUTOMATIC IMAGE ANNOTATION

L. Seneviratne and E. Izquierdo

Multimedia and Vision Research Group, School of Electronic Engineering and Computer Science,
Queen Mary, University of London,
Mile End Road, London, E1 4NS, UK.
phone: +44(0)2078827880, fax: + 44(0)2078827997, email:{lasantha.s, ebroul.izquierdo}@elec.qmul.ac.uk

ABSTRACT

In this paper, an interactive approach to obtain semantic annotations for images is presented. The proposed approach aims at what millions of single, online and cooperative gamers are keen to do, enjoy themselves in a competitive environment. It focuses on computer gaming and the use of humans in a widely distributed fashion. This approach deviates from the conventional "content-based image retrieval (CBIR)" paradigm favoured by the research community to tackle the problems related to the semantic annotation and tagging of multimedia contents. The proposed approach uses a multifaceted mathematical model based on game theories to aggregate numbers of different key-paradigms, such as Image Processing, Machine Learning and Game based approaches to generate accurate annotations. As a consequence, this approach is capable of identifying less-rational (cheating oriented) players, thus eliminating them from generating incorrect annotations. The performance of the proposed framework is tested with a number of game players. Result shows that this approach is capable of obtaining correct annotations in practice.

Index Terms— Semantic annotation, Interactive gaming, MPEG-7 features and object recognition

1. INTRODUCTION

Object recognition and semantic image representations are predominant research topics in the computer vision community. Although the technological developments in recent years for mapping low-level features with high-level concepts have improved, the "semantic gap" still remains as an open challenge in the computer vision community.

Over the last decade, challenges in the semantic gap have attracted researchers from different communities. As a result, a large number of approaches for image annotation have been developed. One such approach is to accomplish image annotation by using collaborative efforts. Collaborative annotation aims at splitting an activity into reasonable chunks to be divided among people who are willing to contribute their resources or efforts [1]. Another approach is to design a user-centric interactive framework that is instrumental in harvesting human intelligence. The "ESP game" [2] and "Manhattan Story Mashup" [3] are two different innovative game strategies that are instrumental in harvesting human brainpower for annotating images. Since humans can inherently describe image semantics the games exploit human cognitive intelligence to annotate images. Furthermore, this activity is a hidden activity and users often do not realize the contribution they have made while playing the games.

The 'Entertainment Software Association', recently illustrated that in the United States alone there are more than 200 million hours spent each day playing computer and video games [4]. Moreover, it statistically shows that by the age of 21 an average American will spend more than 10,000 hours of playing computer games which is equivalent to five years of full time working. What if this time and effort can be utilized to address the semantic gap issue in the computer-vision community? By considering computer gaming and computer vision techniques we believe that there are numerous techniques available to overcome the issue of manual image annotation. One of the most effective ways to overcome this issue is to design interactive frameworks which would be able to captivate a large number of game players. This has to be carefully done by considering the player's psychological aspects; in general, players won't play games for unravelling computational problems, but they do so to entertain themselves.

The remainder of this paper is organized as follows; Section 2 gives an overview of the proposed approach. Section 3 introduces the proposed outcome prediction approach. In section 4, the Two-player game model and Payoff calculation are introduced. In section 5, the evaluation of gaming environment is discussed. Finally, section 6 summarizes the paper along with the future research goals.

2. PROPOSED APPROACH

A diagrammatic overview of the proposed approach is given in Figure 1. The complete framework comprises two modules. The first module (right) analyses the annotations made for fully annotated images and second module (left) analyses the annotations made for partially annaotated and non-annotated images. Hence, the entire framework consists of three databases, namely: fully annotated, partially annotated and non-annotated images. The first module i.e., analyzing fully annotated content, is used to understand player's behaviour, confirm results from statistical inference, as well as, estimate model parameters and the shape of its payoff functions. The second module i.e., analyzing partially and non-annotated content, is the actual annotation engine providing semantic metadata for partially annotated and non-annotated images. The image

subject for annotation is visualized by the visual game interface, where the players are expected to comment on them using a single character string. More details about the interface are given in [5].

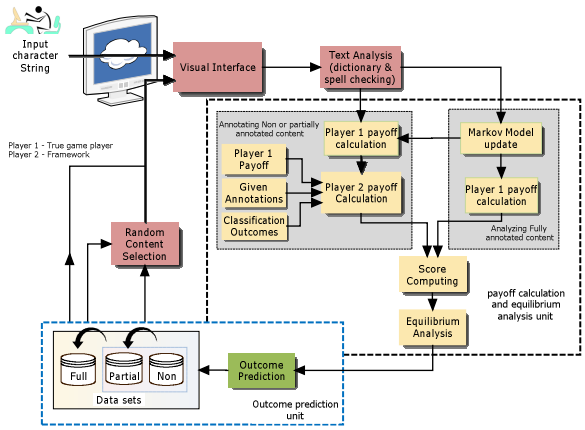

Figure 1 – A complete block diagram of the framework.

At the start, a small set of fully annotated content is fed to the game to initiate the process of learning player's behaviour and model parameters. Next, content is extracted from one of the three available databases (fully, partially or non-annotated) and uploaded into the system. Database selection for content extraction depends on the predicted player's behaviour. However, a module to force extraction of content from the fully annotated multimedia database at random time intervals is also used. The payoff calculation is used to aggregate all the various information. Here, equilibrium analysis [6] provides valuable output on player confidence, i.e., it makes the decision whether to accept a player annotation as correct or incorrect based on a fair trade solution. Score computation is used to motivate players by giving them points thus acknowledging their contribution.

3. PREDICTION OF PLAYER'S BEHAVIOR

There are players with different attitudes, cheating oriented and rational minded. It is always difficult to correctly distinguish all players. In addressing this problem, we have proposed an approach based on sequential sampling plans to predict the player's outcome. This approach uses an Operating Characteristic curve (OC) to demonstrate the player's distribution in gaming; hence it represents the picture of the sampling plan. As shown in Equation (1), player's distribution in gaming is measured by Binomial distribution. It shows exactly x defective annotations in n images as a probability distribution.

$$P(X) = \binom{n}{x} p_{oc}^{x} (1 - p_{oc})^{n-x} \quad (1)$$

Where, variable p_{oc} represents the proportion of non-confirming outcomes (bad annotations) of the incoming annotations. In the proposed sampling plan, acceptance quality (AQL) is measured by (2) based on the quality in detecting correct keywords by the framework, i.e., using the dictionary mechanism. In practice, there is a risk that player's annotation could be rejected by the framework. This could happen when the dictionary mechanism fails in detecting an existing word in the English language. The risk of a correct annotation is being rejected is the risk that player faces in this game and is denoted by α. This risk is measured by the OC curve and its associated AQL parameter.

$$AQL = 1 - \left(\frac{N_c}{N_p}\right) \quad (2)$$

Where, N_c denotes the number of valid labels, i.e., annotations with correct spelling which are correctly detected by the framework, and N_p denotes the number of correct labels given by all players. Rejectable quality level (RQL) is measured by (3) based on the player's commitment to the game. Here, β denotes the probability of accepting a lot of the RQL quality.

$$RQL = \left(\frac{N_w}{N_a}\right) \quad (3)$$

Where, N_w denotes the number of wrong annotations given by the player and N_a denotes the number of fully annotated contents exposed to the player. Figure 2 shows a picture of the proposed OC curve. Here, RQL is frequently updated whenever player annotates a fully annotated content.

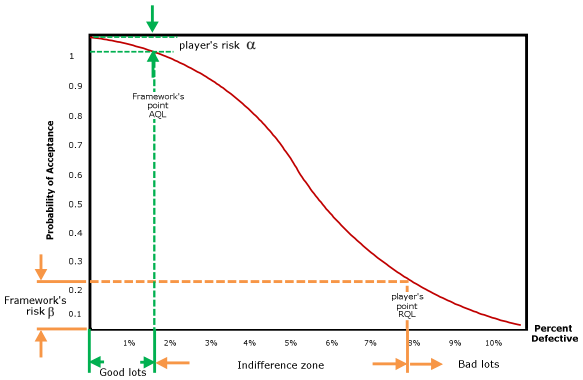

Figure 2 - Operating characteristic curve.

In the proposed approach, item-by-item sequential sampling is used. Here, hitting or crossing a line results in making a decision [7]. Given a set of quality levels, AQL (p_1), α, RQL (p_2), β and n (number of exposed images), the acceptance and rejection lines are calculated by (4) and (5).

$$\text{Acceptance line:} \quad Y_a = -h_1 + Sn \quad (4)$$

$$\text{Rejection line:} \quad Y_r = h_2 + Sn \quad (5)$$

The origin of acceptance line is computed as:

$$h_1 = \frac{\log \frac{1-\alpha}{\beta}}{k_1}$$

The origin of rejection line is computed as:

$$h_2 = \frac{\log \frac{1-\beta}{\alpha}}{k_1}$$

The line slope is computed as:

$$S = \frac{\log\frac{[1-p_1]}{[1-p_2]}}{k_1}$$

Where,

$$k_1 = \log\frac{p_2(1-p_1)}{p_1(1-p_2)}$$

The proposed prediction mechanism works as follows. Before exposing a non-annotated content, number of defective annotations and exposed images are increased by 1. This simulates the worst outcome in this game, which is of having a wrong annotation. Next, the OC curve and other parameters such as, RQL, β, α, Y_a, Y_r and the plotted point in the sequential sampling plan is updated. In this instant, as shown in Figure 3 if the plotted point falls on or below the lower line i.e., acceptance line, a non-annotated content is exposed to the player when only the average good contribution of player $P\ (C) < T_1$, or else, a partially annotated content will be exposed. If the plotted point falls within the parallel lines or above the upper line, a fully annotated content is exposed to the player. The above process will keep exposing fully annotated contents until the lot has been accepted.

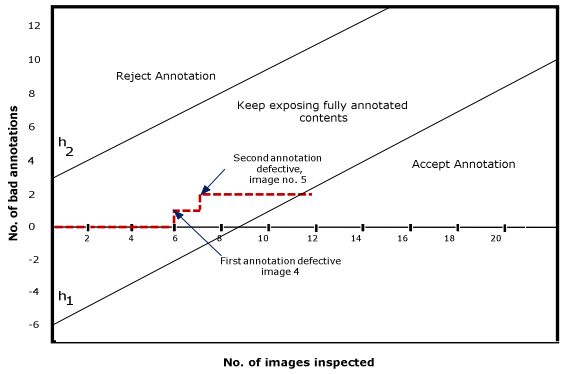

Figure 3 - Proposed sequential sampling plan.

4. TWO-PLAYER GAME MODEL AND PAYOFF CALCULATION

Initially the framework feeds players with a number of fully annotated images; then it analyzes all the annotations in order to measure player confidence, thus, the transition probabilities. This is done by using a Markovian model [8]. The two states of the Markov Model (MM) are: a "correct" and an "incorrect" tag or annotation is entered, and they are represented by the variable C and I, respectively. The outcomes for fully annotated contents are sequentially ordered and segmented into sets of tags for the purpose of calculating conditional probabilities in the transition matrix. For example, the probability of $P(C_{t+1}|I_t)$ is estimated by dividing the number of sets in which the label 'correct' occurs before 'incorrect' by the total number of tag sets containing 'incorrect'. In Figure 4, an overview of the segmenting process is given.

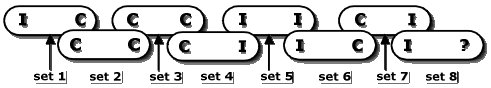

Figure 4 - Segmenting player's outcome into set of tags

Given by the player outcome at time t on preceding multimedia content, the probabilistic outcome at time $t+1$ is estimated by using the transition matrix M. This matrix gives the change of behaviors of player in the Markovian chain.

$$M = \begin{bmatrix} P(C_{t+1}|C_t) & P(I_{t+1}|C_t) \\ P(C_{t+1}|I_t) & P(I_{t+1}|I_t) \end{bmatrix}$$

Where, $P(C_{t+1}|C_t)$ denotes the probability of obtaining a correct annotation at time $t+1$, when player has given a correct annotation at time t. Similarly, other probabilities $P(C_{t+1}|I_t), P(I_{t+1}|C_t)$ and $P(I_{t+1}|I_t)$ are measured using player's historical data, i.e., using segmented outcomes. A diagrammatic overview of the proposed MM is given in Figure 5.

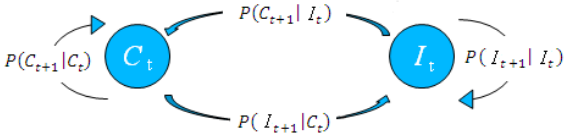

Figure 5 - Player's probability distribution in gaming.

Since players do not know as to what type of content they are exposed to, it is sensible to assume that they respond in the same way to any of the three types of content: fully annotated, partially annotated or non-annotated. Using this assumption, player 1's payoff is always estimated by calculating overall good (G_i) and bad (B_i) contributions in gaming.

$$P(G_1) = \left(P(C_{t+1}|C_t)P(C) + P(C_{t+1}|I_t)P(I)\right)$$

Where, $P(C_{t+1}|C_t)P(C)$ is the probability of having correct annotations at $t+1$, when the state 'correct' considered and $P(C_{t+1}|I_t)P(I)$ is the probability of having correct annotations at time $t+1$, when the state 'incorrect' considered.

$$P(B_1) = \left(P(I_{t+1}|C_t)P(C) + P(I_{t+1}|W_t)P(I)\right)$$

Where, $(I_{t+1}|C_t)P(C)$ is the probability of having incorrect annotations at $t+1$ when the state 'correct' considered and $P(I_{t+1}|I_t)P(I)$ is the probability of having incorrect annotations at $t+1$ when the state 'incorrect' considered. When player 2 is considered, $P(G_2)$ in gaming is estimated as,

$$P(G_2) = \left(\pi_1(a_1, a_2) + P(K) + P(F)\right)/k_2$$

$$k_2 = \begin{cases} 3, & \text{if } P(K) \text{and } P(F) \text{ available} \\ 2, & \text{if } P(K) \text{ or } P(F) \text{ available} \\ 1, & \text{otherwise} \end{cases}$$

Where, $\pi_1(a_1, a_2)$ is the overall payoff of player 1 in gaming; $P(K)$ is the probability of entering a given annotation, i.e., number of annotations given similar to the player input keyword / total number of annotations; $P(F)$ is the outcome of low-level feature classification [9]; k_2 is the normalising constant that defines the availability of $P(K)$ and $P(F)$. In practice, classification outcomes are not entirely accurate and therefore, are being used when only classification outcomes are greater than the F-measure of the concept. When considering player 2, $P(B_2)$ in gaming is estimated as:

$$P(B_2) = N * T_2$$

Where, N is the number of dissimilar annotations assigned to an image (measured by the Wordnet dictionary tool) and T_2 is the allocated cost per annotation. If the framework performs good in annotation, it can be assumed that the number of N would be smaller.

When we assume that player is given a non- or partially annotated content, the profile of actions is estimated as follows. Let the action of player i taken at each round be a_i. Action a_1 indicates that outcome of player 1 is good or bad in a game round and is observed by the output prediction module. When the outcomes of prediction say player will enter a good annotation, it is assigned $a_1 = 1$.

$$a_1 = \begin{cases} 1, \text{if prediction says good annotation} \\ 0, \text{ otherwise} \end{cases}$$

a_2 is the player 2's action property, and is being calculated using a threshold score. When player 1's game score is less or equal to a certain threshold score T_3 (*player 1 score* ≤ *threshold score*), action a_2 is assigned 0. a_2 is assigned 1 when the player score is greater than the threshold score (*player 1 score* > *threshold score*). Although player 1 increases his score by feeding in 'correct' annotations, framework keeps a difference in game points between the player's score and threshold score. Whenever player cheats, his/her score will be reduced according to the score computation module, while keeping the threshold score unchanged. Additionally, whenever player 1's score is less than the threshold score, the threshold score will be kept unchanged until the player score becomes greater than the threshold score with a lead of T_3. Therefore, it can be assumed that this process represents the long term contribution of the player in gaming.

$$a_2 = \begin{cases} 1, \text{ if } player\ 1\ score > threshold\ score \\ 0, \text{ otherwise} \end{cases}$$

For each round, given all information including action profile (a_1, a_2), a general function for calculating player 1 and 2's payoff can be defined by (6) and (7).

Payoff of player 1:

$$\pi_1(a_1, a_2) = a_1\, P(G_1) - a_2\, P(B_1) \qquad (6)$$

Payoff of player 2:

$$\pi_2(a_1, a_2) = a_2\, P(G_2) - a_1 P(B_2) \qquad (7)$$

Table 1 - Payoff representation for all actions.

Actions		Player 2's long term contribution level	
		bad (a_2= 0)	good (a_2= 1)
Player 1's short term contribution	bad (a_1= 0)	(0, 0)	(-, +)
	good (a_1= 1)	(+, 0)	(+, +)

If players are cooperative, Table 1 shows action pair *Short good, Long good* forms the unique Nash equilibrium. It can be simply found by analyzing game outcomes for each action configuration. Score computation module is used for two purposes; firstly, to reward players for their contribution in gaming thus, to yield game points; secondly, to measure action property of the player 2 (a_2).

$$Player\ i's\ score = player\ i's\ payoff * 100$$

5. PERFORMANCE MEASURE

The performance of the proposed framework is studied for 96 fully annotated, 48 partially and 60 non annotated images. All experiments were conducted with the following threshold parameters, which have been chosen using a validation set of images, not part of the test set that used to measure performances of the proposed model. The threshold T_1, i.e., exposing partially or non-annotated content is assigned a value of 0.63. It has been found that the good contribution of true game players are always greater than 0.63. By assigning this number partially annotated contents will be mostly exposed to the true game players. As a consequence, more accurate annotations are extracted. AQL is assigned a value of 0.03; threshold T_2 (limits the maximum number of annotations per content) is assigned a value of 0.166. In practice, it has been found that minimum of 1 and maximum of 5 annotations are needed to find a single correct annotation. By assigning 0.166, framework allows players to annotate images with 5 different annotations. Threshold T_3 (used for a_2 calculation) is assigned a value of 301. This is an acceptable number for the game because it has been found that rational minded players do not complete 3 incorrect annotations in a single row.

In Figure 8, a correct annotation detected by the framework i.e., true positive, is shown by a square sign; an incorrect annotation completed by the framework i.e., false positive, is shown by a triangular.

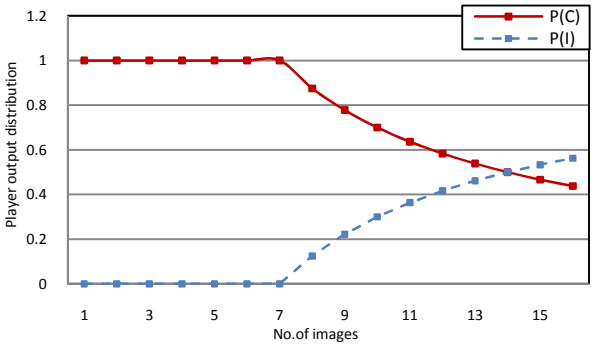

Figure 6 - Performance of a classical player.

Figure 7 - Performance of a random player.

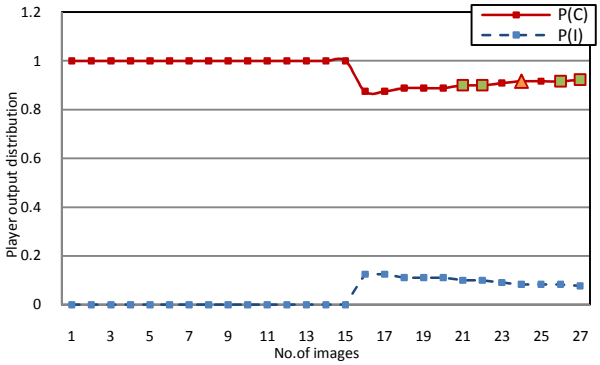

Figure 8 - Performance of a true player.

The proposed framework is evaluated using 310 human players. Figures 6-8 show the framework outcomes for a selected classical (a player who does good annotations in the beginning of the game and then cheats), random and genuine player. For classical cheaters, the overall accuracy of the system is 84% and for random cheaters it was about 79%. For true game players the accuracy is about 89%. This leads to an overall accuracy of the system for 84% in image annotation.

Figure 9 shows comparison results of the proposed method against a conventional framework, i.e., a framework where no mathematical techniques are involved and as a fact it collects all the annotations that are given by the player, and another framework which is proposed in [9] based on a Markovian model. Here, some modifications have been done to the Markovian based system to improve its performances.

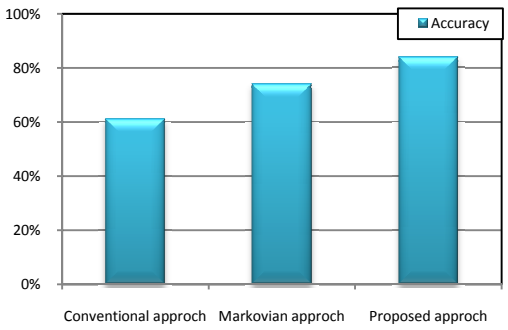

Figure 9 - Comparison of different frameworks.

It shows in terms of accuracy, the proposed framework outperforms both other frameworks in image annotation.

6. CONCLUSIONS AND FUTURE WORK

We have presented a labour-intensive approach to harvest human brain power for addressing the semantic gap problem in the computer vision community. The proposed approach is a standalone game that is capable of entertaining, motivating and is used to provide valuable information on image contents. Besides the fun factor, this framework provides high accuracy in image annotation. Result shows that the proposed framework outperforms Markovian based framework and the conventional labour-intensive framework. This approach extended the behaviour of a conventional labour-intensive game by eliminating annotations from less-rational gamers. As a result, accuracy in image annotation is significantly improved.

Future research will focus mainly on improving frameworks efficiency, and to obtain a large number of annotations using a small numbers of players.

REFERENCES

[1] L. Kristina and A. Laurie, "Social Browsing on Flickr," in *Fifth International AAAI Conference on Weblogs and Social Media*, 2007.

[2] L. von Ahn and L. Dabbish, "Labelling images with a computer game," in *proceedings of the ACM Conference on Human Factors in Computing Systems (CHI)*, 2004.

[3] V. Tuulos, J. Scheible, and H. Nyholm, "Combining Web, Mobile Phones and Public Displays in Large-Scale: Manhattan Story Mashup," in *proceedings of the 5th international conference on Pervasive computing*, 2007, pp. 37-54.

[4] L. von Ahn and L. Dabbish, "Designing games with a purpose," *Communications of the ACM*, vol. 51, no. 8, 2008.

[5] L. Seneviratne and E. Izquierdo, "Image annotation through gaming (TAG4FUN)," in *16th International Conference on Digital Signal Processing*, 2009.

[6] G. Scutari, D. P. Palomar, and S. Barbarossa, "Optimal Linear Precoding Strategies for Wideband Noncooperative Systems Based on Game Theory-Part I: Nash Equilibria," in *IEEE Transactions on Signal Processing, vol. 56, no. 3,* 2008, pp. 1230-1249.

[7] A. K. M. Abdul and F. A. Burney, "Program for Item-by-Item Sequential Sampling by Attributes," King Abdulaziz University, Technical Report 1992.

[8] M. L. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming.*: John Wiley & Sons, Inc, 1998.

[9] L. Seneviratne and E. Izquierdo, "An interactive framework for image annotation through gaming," in *Proceedings of the 11th ACM SIGMM International Conference on Multimedia Information Retrieval*, 2010.