Information Fusion in HMM-Based Visual-Task Inference

Anonymous

Abstract

The effect of visual task on the pattern and parameters of eye movements has been long investigated in the oculomotor studies of human vision. However, there is not much done in the inverse process; that is inferring the visual task from eye movements. Visual search is one of the main ingredients of human vision that plays an important role in our everyday life. In this paper, we develop a probabilistic framework to infer the ongoing task in visual search given the eye movements. We propose to use a dynamic programming method called *token passing* in an eye-typing application to reveal what the subject is typing during a search process by observing his direction of gaze during the execution of the task. Token passing method is a computationally simple technique that allows us to fuse higher order constraints in the inference process. In the experiments we examine the effect of higher order information, in the form of a lexicon dictionary, on the task recognition accuracy.

Introduction

Human vision is an active, dynamic process in which the viewer seeks out specific visual inputs according to the ongoing cognitive and behavioral activity. A critical aspect of active vision is directing a spatially circumscribed region of the visual field (about 4^o) corresponding to the highest resolution region of the retina, the so-called fovea, to task-relevant stimuli in the environment. This way our brains will get a clear view of the conspicuous locations in an image and will be able to build up an internal, task-specific, representation of the scene.

In visual activities the eyes make rapid movements, called *saccades*, typically between two and five times per second in order to bring environmental information into the fovea. Pattern information is only acquired during periods of relative gaze stability, called *fixations*, owing to the brain's suppression of information during the saccades (Matin 1974). Gaze control, thus, is the process of directing the fovea through a scene in real time in the service of ongoing perceptual, cognitive and behavioral activity. The question of exactly what is happening during fixations is still something of a puzzle, but the effect of visual task on the pattern and specifications of eye movements has been long studied in the literature.

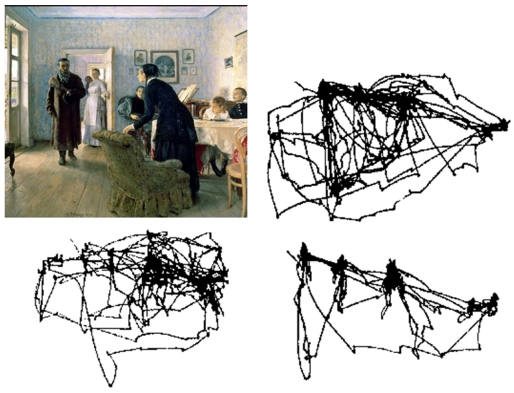


Figure 1: Eye trajectories measured by Yarbus by viewers carrying out different tasks. Upper right - no specific task, lower left - estimate the wealth of the family, lower right - give the ages of the people in the painting (Yarbus 1967).

In two seminal studies, Yarbus (Yarbus 1967) and Busswell (Buswell 1935) showed that visual task has a great influence on specific parameters of eye movement control. Figure 1 shows Yarbus's observation that eye fixations are not randomly distributed in a scene but instead tend to cluster on some regions at the expense of others. In this figure we can see how visual task can modulate the conspicuity of different regions and as a result change the pattern of eye movements.

Although the effect of visual task on eye movement pattern has been investigated for various tasks, there is not much done in the area of visual task inference from eye movements. In other words, in a forward Yarbus process the visual task is given as an input and the output is task-dependent scanpaths of eye movements. In this work, on the other hand, we develop a method to realize an inverse Yarbus process whereby we can infer the ongoing task by observing eye movements of the viewer.

Visual search is one of the main ingredients of many complex tasks. When we are looking for a face in a crowd or counting the number of certain objects in a cluttered scene,

we are unconsciously performing visual search to look for certain features in the faces or in the objects in a scene. As proposed in (Treisman and Gelade 1980), the level of difficulty in a search task can vary according to the number of features distinguishing the target object from the distractors. For instance, targets defined by a unique color or a unique orientation are found more easily compared to the ones defined by a conjunction of features (e.g. red vertical bars). On this basis, we have two types of visual search called *pop-out search* and *conjunction search*. In pop-out search, the complexity of determining the presence of a target, measured by reaction time (RT), is independent of the number of distractors in the scene, while in the conjunction search it is highly number-of-distractor-dependent (Treisman and Gelade 1980).

Recently, Haji-Abolhassani and Clark (Haji-Abolhassani and Clark 2011) have implemented a model based on the theory of hidden Markov models (HMMs) to infer the visual task in pop-out visual search. The tasks used there were to look for certain objects in computer-generated stimuli that could be distinguished by a single feature (e.g., searching for red bars) and the inference was made by classifying the test data into one of the 6 classes of pre-defined, pop-out tasks. Moreover, a uniform distribution was used as the prior information about the tasks leading to a *maximum likelihood* (ML) estimation of the task.

In this paper we extend the HMM-based model of Haji-Abolhassani and Clark (Haji-Abolhassani and Clark 2011), called single-state HMM (SSHMM), to infer the task in conjunction visual search, which can potentially take up an unlimited number of tasks as the task set resulting in recognizing the task rather than classifying it. Moreover, in real life tasks don't happen according to a uniform distribution (as assumed in the ML estimator of SSHMM) and have different a-priori probabilities. Therefore, in order to infer the ongoing task, we propose to use our HMM models within a simple conceptual model of eye-movement recognition based on a technique called *token passing* that incorporates the sub-task HMMs in a transition network structure. The higher order constraints are, then, applied to the recognition along transitions from an HMM unit to another.

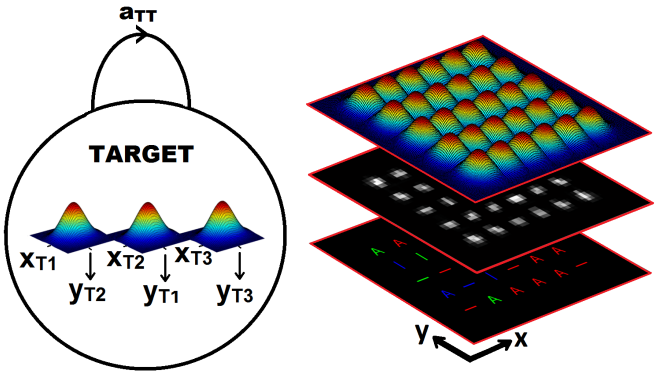
In the following sections we will first revisit the SSHMM model developed in (Haji-Abolhassani and Clark 2011) for task inference in pop-out search. Knowing the architecture of the task inference model, we extend it to task inference in conjunction search and equip it with high level constraints and see how it can improve the recognition rate.

Task Inference in Pop-Out Search

Since the pop-out search is distractor-independent (Treisman and Gelade 1980), attention is mostly directed to task-relevant objects in the scene. Therefore, in the SSHMM, attention is assumed to be mostly on targets and the conventional structure of HMMs is degenerated to a single-state, self returning one; hence the name single-state HMM (SSHMM). In the generic structure of the SSHMM, the state represents the attention demanding targets in each task.

While it is well known that there is a strong link between eye movements and attention (Rizzolatti, Riggio, and Sheliga 1994), the attentional focus is nevertheless frequently well away from the current eye position (Fischer and Weber 1993). Eye tracking methods may be appropriate when the subject is carrying out a task that requires foveation. However, these methods are of little use (and even counter-productive) when the subject is engaged in tasks requiring peripheral vigilance. Moreover, due to the noisy nature of eye-tracking equipment, the actual eye position itself is usually different from what the eye-tracker shows, which will bring in systematic error to the estimations.

Therefore, For the target locations it is postulated that the observations are random outcomes of a mixture of 2-D Gaussians with features x and y in a Cartesian coordinates that are maximum on the centroids of the targets and fade away as we become more distant (Euclidian) from them (see figure 2a). In figure 2b we have put the observation pdfs of all the objects together and have superimposed them on the original image and its corresponding bottom-up saliency map (Itti and Koch 2001). It is from this grid of Gaussians that we select the ones related to the task and combine them into a Gaussian mixture model (GMM) to represent the state's observation pdf.



(a) (b)

Figure 2: a) Generic SSHMM for task inference in pop-out visual search scheme as proposed in (Haji-Abolhassani and Clark 2011). The transition matrix is composed of a deterministic loop from the state to itself and the observation pdf comprises mixture of Gaussians centered around target-relevant objects in the image (T1, T2, T3, etc.). b) Observation pdfs give us the probability of seeing an observation given a hidden state. In this figure we have put the fixation location pdfs of all the targets together and superimposed them on the original image and its corresponding bottom-up saliency map.

Task Inference in Conjunction Search

In SSHMM the idea of using HMMs is successfully applied to infer the ongoing task in pop-out visual search where the targets differed from the surrounding distractors by a unique visual feature; such as color, orientation, size or shape; and could be located in a stimulus within a short period of time. Nevertheless, in real-life we usually encounter situations

Figure 3: The schematic of the on-screen keyboard used in the experiments. We have removed "Z" in order to have a square layout to reduce directional bias. Also the location of each character is randomized in each layout so that the user has to search for the characters.

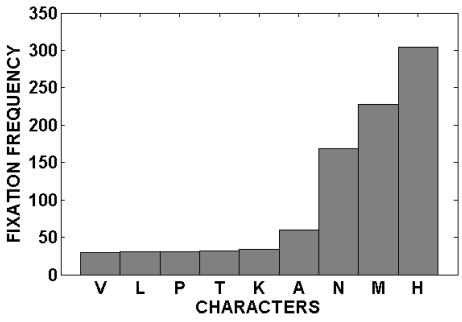
where the target is surrounded by distractors with similar features and can be distinguished from them only by comparing a combination of visual features. Here, we extend the method to a more complicated group of tasks and investigate the applicability of the HMM-based method in task inference in conjunction visual search. Moreover, the experiment for the pop-out search was done on synthetic stimuli with a limited set of simple tasks whereas in the new model we use a more realistic application to evaluate our proposed model on a wider range of tasks.

Searching for a character among the others is investigated in the literature related to visual search and is shown to be massively serial and self-terminating, which suggests that the long-term familiarity with letters do not eliminate the conjunction effect (Treisman and Gelade 1980). Thus, in this paper we investigate task inference in conjunction search by developing an eye-typing application where users can type a character string by directing their gaze through an on-screen keyboard (see figure 3). In this scenario inferring the task is equivalent to figuring out what word has been eye-typed by seeing the eye movements of the subject while performing the task; hence wide range of tasks. In order to force visual search, we randomized the location of characters in the keyboard layout in each trial .

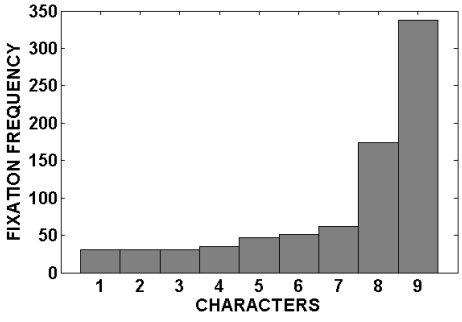
In conjunction search a combination of features is used to define the targets. This characteristic calls for an attentive, mainly serial, limited capacity attentional deployment over a limited portion of the visual field as opposed to a massively parallel attentional deployment across large portions of the visual field in pop-out search (Wolfe 1994; Wolfe, Cave, and Franzel 1989; Treisman and Gelade 1980). The serial nature of this type of search, thus, will multiply the reaction time (RT) when the number of distractors rise and consequently will cause several *off-target fixations* on non-target objects to examine their task relevant features and dismiss them from potential target locations. These off-target fixations on objects that are not directly relevant to the ongoing task are not fully inline with the structure of the SSHMM proposed for pop-out search and will presumably cause an attenuation in the accuracy of task inference.

In the new model we tailor the SSHMM structure to allow for off-target fixations that are very common in conjunction search paradigm. We propose to add another state to represent the off-target fixations. Moreover, we believe even off-target fixations carry information about the sought target. Figure 4a shows the top nine bars of the histogram of fixations on characters (fixation distribution) when looking for character "W". It can be seen that even off-target fixations show a pattern in the sense that seemingly similar characters tend to draw attention towards themselves more often than dissimilar ones. Similar results were obtained when analyzing other characters, too.

This phenomenon is well studied before in the psychological literature related to perceptual measurement of image similarity (Keren and Baggen 1981). Our finding is inline with a psychophysical experiment in (Gilmore et al. 1979, Figure 1) that categorizes uppercase English letters according to their similarity in appearance. Figure 4b shows the average of top 9 fixation location histogram bars when looking for different characters. This trend suggests that off-target fixations can also be used as another source of information. Namely, when looking for a target, similar characters are more likely to be found among the off-target fixations which can help us narrow down our choices in the inference process.



(a)



(b)

Figure 4: Spatial distribution of fixations (fixation distribution) while searching for a character. a) shows the top nine bars of the fixation distribution when looking for character "W". Seemingly similar characters tend to draw attention towards themselves which is inline with the psychological experiments (Gilmore et al. 1979). b) shows the average of top 9 fixation location histogram bars when looking for different characters in the keyboard.

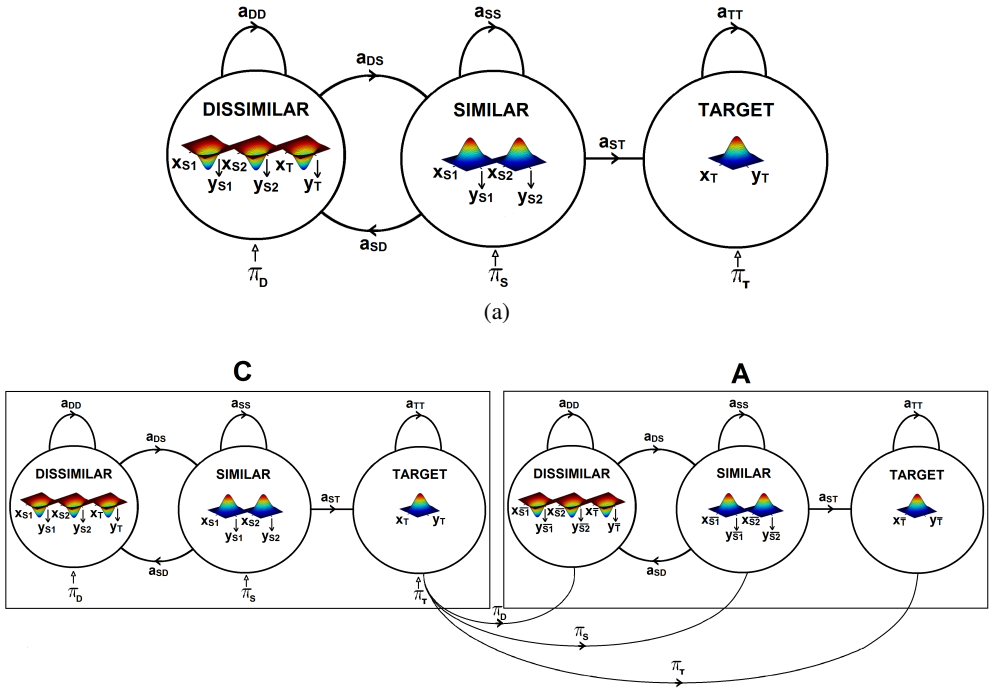


Figure 5: The structure of tri-state HMM (TSHMM) for character recognition. a) shows the TSHMM for a single character. The mean vector of the S-state GMM points to the top two characters of the fixation distribution of the training data of the target character. b) shows how to concatenate the character models to build up the word model. The transitions between the states are governed by the initial state probabilities.

Figure 5a shows the new structure we propose to be used as the generative model of eye movements when looking for a character. In this new setup, we split the off-target state to *dissimilar state* (D-state) and *similar state* (S-state) according to the similarity of the off-target fixation to the target character. We believe the new tri-state HMM (TSHMM) for character recognition allows us to investigate the fixations in more detail and gives us more a-posteriori information given the eye movements. In this model not only the dynamics of on-target fixations are taken into account, but also the off-target fixations play an important role in revealing the target character. Figure 5b shows how we build a word model by concatenating the HMMs of the comprising characters. As can be seen, the transitions between the characters are made from the target state. We heuristically select the top two characters of the fixation distribution of each target to model the GMM of its S-state. The distribution function of the D-state is obtained by complementing the mixture of target Gaussian pdf and GMM of the S-state (all with the same weight). When going from one character to another, the transition probabilities are assumed to be proportional to the initial state probabilities.

Information Fusion Using Token Passing

Although the structure of TSHMMs is more compatible with the nature of conjunction search resulting in potentially better results in task inference, there are other sources of information that could be applied to the inference to improve the performance of the model. Probability distribution of task priors is a source of information that we use on a daily basis to make inferences about our observations. In our application, when the model gives us a uniform a-posteriori distribution over characters "V" and "U", knowing that the proceeding character was a "Q" would help us choose "U" as the eye-typed character, because that is the character that always follows "Q" in common English words.

A similar technique is used in speech processing literature to improve the result of a recognizer by applying high level constraints to the character sequences (Rabiner 1990). The constraint is imposed to the decision making engine in the form of a lexicon dictionary called *language model* that provides us with prior probabilities of seeing different characters given the current one.

Since in our application we are dealing with common English words as well, we use a similar technique to apply higher order constraints on the recognizer. In order to build the language model (LM) we need to get a database of valid English words. Then we can train the LM by assuming a first order Markov chain as the underlying process of character sequences. The training is done by counting the number of each pair of transitions in the corpus. In the end a technique called *add one smoothing* is applied to the count numbers by assuming each pair occurs once more than it actually does to assign non-zero probabilities to the unseen pairs in the train-

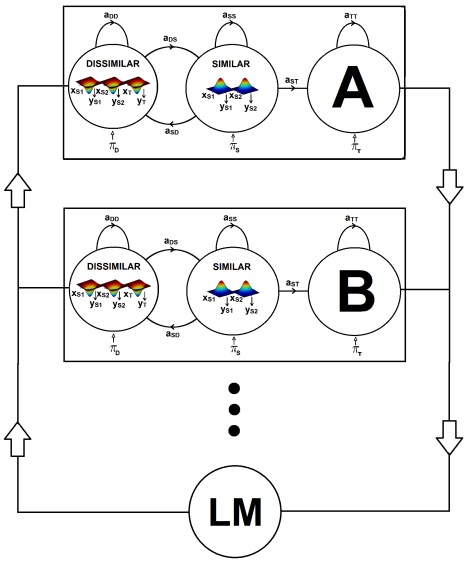


Figure 6: The word lexicon for three-letter words. The best state sequence for a given observation sequence can be obtained by using the token passing technique on this general word model. The circles at the bottom is where language model parameters are applied to the transitions.

ing corpus (Huang et al. 2001, chapter 11). Eventually the language model will give us the probability of p_{ij} for each pair of characters (i, j).

In the previous section we showed how we can train TSHMMs for each character. Therefore, by training the LM we have a complete model for the words in the dictionary that describes the transitions within the states of characters, as well as transitions between a word's characters, in a probabilistic manner. This model can be used as a cognitive model of human brain that generates eye movements during visual search for characters of a word (i.e., eye-typing). First we start from the initial state of a character according to the initial state probabilities of the HMM, and by following the transition probabilities we can choose the states for each time step and generate observations according to the observation probabilities. When getting to the final state of a character, it is the language model that suggests which character, by what probability, can follow the current one. The complete structure of the model is shown in figure 6.

Having the generative model of eye movements during visual search, we can use the trained parameters to decode a test eye movement trajectory to infer what character sequence has been eye-typed. If we had a limited number of hypotheses (words in the dictionary), we could use *Viterbi algorithm* to classify the test date into one of the words in the dictionary (Rabiner 1990). Viterbi algorithm, though, requires the word models to be built beforehand to be able to compare the likelihood of each word in the dictionary. However, for a recognition task there might be an enormous number of words in the dictionary which makes it very expensive to build the word model for each word and therefore renders the Viterbi algorithm computationally expensive.

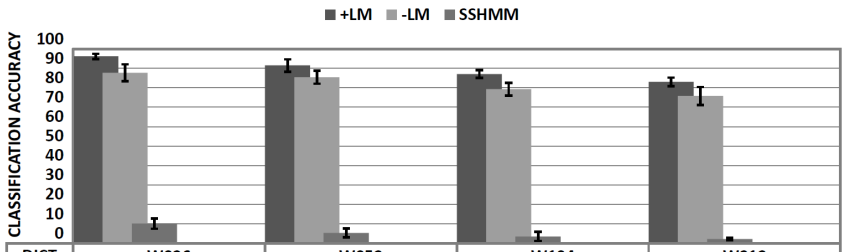
Algorithm 1 Token Passing

Initialize:
Assign a zero valued token to the initial states of the word models.
Assign an infinity valued token to all other states.
Algorithm:
for t:=1 to T **do**
  for each state i **do**
    Copy the token in each state i to the connecting state j and increment its value by $1/p_{ij} + 1/d_j(t)$
  end for
  Discard the original tokens.
  for each state i **do**
    Keep the token with the minimum value and discard the rest.
  end for
end for
Termination:
In final states, the token with the smallest value corresponds to the best match.

An analogous problem exists in the literature related to speech recognition, where the dictionary of possible words exceeds a certain number. The technique used there, that we propose to be used for our problem as well, is an algorithm called *token passing* (Young, Russell, and Thornton 1989). In order to find the best sequence of states that matches the observation sequence we assign a reward to each transition from state i to j equal to p_{ij} and call it *transition reward*. p_{ij} is defined by the parameters of the TSHMMs and the language model. If the transition is within a character model, transition matrix of the TSHMM of the character defines the reward and otherwise, LM statistics should be used to evaluate the transition reward. The second type of reward is called *local reward function* and defines the reward of being at state j at time t. The local reward $d_j(t)$ is the corresponding observation probability of TSHMMs since the definition implies the generation of observation from state j. Algorithm 1 shows how we can use these rewards to decode a sequence of eye movements by finding the most rewarding path in figure 6.

Experiments

To build a database of task-dependent eye trajectories, we ran a set of trials and recorded the eye movements of six subjects while eye-typing 26 3-character words. The trials started with a fixation mark of size 0.26×0.26 deg appearing at the center of the screen. After foveating the fixation mark, the participant initiated the trial with a key-press. Once a trial was triggered, the word to be eye-typed was shown at the center of the display. Once the subject indicated his readiness by pressing a key, another fixation mark appeared at the center followed by an on-screen keyboard similar to the one shown in figure 3. At this phase subjects eye-typed the word by searching for the characters appearing in it as quickly as possible and signaled when they were done by



	DICT.	W026	W052	W104	W312
MEAN	+LM	96.2	91.5	87.1	83.1
	-LM	87.7	85.4	79.3	75.8
	SSHMM	10.1	5.3	3.5	2.1
SEM	+LM	1.3	3.2	2.1	2.2
	-LM	4.4	3.3	3.2	4.6
	SSHMM	2.6	2.3	2.4	0.6

Figure 7: Comparison of task classification accuracy using TSHMM with LM (+LM), TSCHMM without LM (-LM) and SSHMM methods in conjunction visual search. Each bar demonstrates the mean classification rate (%) of correctly recognizing the intended word in the eye-typing application. The mean value and the standard error of the mean (SEM) are represented by bars and the numerical values are given in the following table.

pressing a key (subjects were only told to eye-type the words as quickly as possible and press a key when done). Then by asking about the location of one of the characters (selected randomly) we verified to see if the subject had correctly eye-typed the words. Once the question is answered (by fixating the right location that contained the character during the experiment and pressing a bottom) the next word is shown and the trial carries on.

The stimuli were generated by a computer and displayed on a 1280×800 pixel screen at a distance of 18 inches (1 degree of visual angle corresponds to 30 pixels, approximately). Each keyboard was composed of 25 uppercase English characters randomly located on a 5×5 grid superimposed on a gray background (we removed "Z" in order to have a square layout to reduce directional bias). The 3-letter words were selected so that there was no repetition of characters in them. At the beginning of every experimental session, we calibrated the eye tracker by having the participant look at a 16-point calibration display that extended to 10×10 degrees of visual angle (the area covered by the calibration grid is stretched beyond the stimuli).

An eye tracker (ISCAN RK-726PCI) was used to record the participant's left eye positions at 60 Hz and a chin rest was used to minimize head movements. The eye tracker's vertical resolution is approximately 0.11 degrees and its horizontal resolution is 0.06 degrees. An LCD monitor was used for displaying the images and the subjects used both eyes to conduct the experiments.

After recording eye movements, data analysis was carried out on each trial wherein we removed the blinks, outliers and trials with wrong answers in the verification phase from the data and classified the eye movement data into saccades and fixations. Moreover, in some of the initial trials, after eye-typing the word, the viewer returned to the locations of the characters to double-check the coordinates of them. In order to simulate a real eye-typing application we removed these parts from the trajectories in the pre-processing, too.

After the preprocessing we obtained a database of 145 trajectories of the form $(O_1, \ldots, O_m)$, each containing observation sequences of coordinates of fixations while performing the eye-typing, where $O_i = (x_i, y_i)$ represents x-coordinate and y-coordinate of the i^{th} fixation, respectively.

In order to perform the evaluation, we compare the results of TSHMM with the one proposed in (Haji-Abolhassani and Clark 2011) (SSHMM) in four different dictionary sizes. We have created four sets of dictionaries of 26, 52, 104 and 312 English words using the Carnegie Mellon pronouncing dictionary (CMPD) (Weide 2005). All dictionaries were built so that they all include all the words of the smaller dictionaries. The words were selected randomly from the CMPD and the words length varied between three to five characters. The language model was also created using the CMU-Cambridge toolkit (Clarkson and Rosenfeld 1997) by extracting language models from the words in dictionaries.

In order to train the TSWHMMs, we have to adjust the mean vector of the 2-D Gaussians according to the training character so that it aligns with the center of character location. According to (Rabiner 1990) a uniform (or random) initial estimation of initial state and transition probabilities (Π and A) is adequate for giving useful re-estimation of these parameters (subject to the stochastic and the nonzero value constraints). Thus, we have set a random initial values for the parameters in the generic HMM and run the *Baum-Welch* algorithm on the training set to obtain the final TSWHMM (Huang, Ariki, and Jack 1990). We have also used a technique called parameter tieing (Rabiner 1990) to force a unique task and stimuli independent covariance matrix across all the Gaussian distributions in the mixtures. Thus, we can build the word model for the test data by dynamically changing the means of the states according to the character locations of the characters and using the estimated variances of characters.

Figure 7 shows the accuracy of word inference using TSWHMM with LM (+LM), TSCHMM without LM (-LM)

and SSHMM methods ranging over four dictionary sizes. As expected, the +LM performs better than -LM due to the fusion of information provided by the LM. Since SSHMM is designed for pop-out search, it fails to infer the task in conjunction search where we have multiple off-target fixations in an average trajectory. The table below the figure shows the accuracy and the standard error of the mean (SEM) of the corresponding bars. For each bar we ran a 10-fold cross validation on our database of 145 trajectories in order to define the training and test sets and used the same epochs across all the methods.

References

Buswell, G. 1935. *How people look at pictures: A study of the psychology of perception in art*. Chicago: University of Chicago Press.

Clarkson, P., and Rosenfeld, R. 1997. Statistical language modeling using the cmu-cambridge toolkit. In *Fifth European Conference on Speech Communication and Technology*.

Fischer, B., and Weber, H. 1993. Express saccades and visual attention. *Behavioral and Brain Sciences* 16:553–553.

Gilmore, G.; Hersh, H.; Caramazza, A.; and Griffin, J. 1979. Multidimensional letter similarity derived from recognition errors. *Attention, Perception, & Psychophysics* 25(5):425–431.

Haji-Abolhassani, A., and Clark, J. 2011. Visual task inference using hidden markov models. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence*. IJCAI/AAAI.

Huang, X.; Ariki, Y.; and Jack, M. 1990. *Hidden Markov Models for Speech Recognition*. Edinburgh University Press.

Huang, X.; Acero, A.; Hon, H.; et al. 2001. *Spoken language processing*. Prentice Hall PTR New Jersey.

Itti, L., and Koch, C. 2001. Computational modelling of visual attention. *Nature Reviews Neuroscience* 2(3):194–204.

Keren, G., and Baggen, S. 1981. Recognition models of alphanumeric characters. *Attention, Perception, & Psychophysics* 29(3):234–246.

Matin, E. 1974. Saccadic suppression: A review and an analysis. *Psychological Bulletin* 81(12):899–917.

Rabiner, L. 1990. A tutorial on hidden Markov models and selected applications in speech recognition. *Readings in speech recognition* 53(3):267–296.

Rizzolatti, G.; Riggio, L.; and Sheliga, B. 1994. Space and selective attention. *Attention and performance XV: Conscious and nonconscious information processing* 231–265.

Treisman, A., and Gelade, G. 1980. A feature-integration theory of attention. *Cognitive psychology* 12(1):97–136.

Weide, R. 2005. The carnegie mellon pronouncing dictionary [cmudict. 0.6].

Wolfe, J.; Cave, K.; and Franzel, S. 1989. Guided search: An alternative to the feature integration model for visual search. *J. Exp. Psychol* 15:419–433.

Wolfe, J. 1994. Guided search 2.0 a revised model of visual search. *Psychonomic bulletin & review* 1(2):202–238.

Yarbus, A. 1967. Eye movements during perception of complex objects. *Eye movements and vision* 7:171–196.

Young, S.; Russell, N.; and Thornton, J. 1989. Token passing: a simple conceptual model for connected speech recognition systems. *Cambridge University Engineering Department* 1–23.