

Provided for non-commercial research and education use.
Not for reproduction, distribution or commercial use.



This article appeared in a journal published by Elsevier. The attached copy is furnished to the author for internal non-commercial research and education use, including for instruction at the authors institution and sharing with colleagues.

Other uses, including reproduction and distribution, or selling or licensing copies, or posting to personal, institutional or third party websites are prohibited.

In most cases authors are permitted to post their version of the article (e.g. in Word or Tex form) to their personal website or institutional repository. Authors requiring further information regarding Elsevier's archiving and manuscript policies are encouraged to visit:

<http://www.elsevier.com/copyright>



Contents lists available at ScienceDirect

Vision Research

journal homepage: www.elsevier.com/locate/visres

Efficient bubbles for visual categorization tasks

Hong Fang Wang^{a,*}, Nial Friel^b, Frederic Gosselin^c, Philippe G. Schyns^a^aInstitute of Neuroscience and Psychology, University of Glasgow, UK^bSchool of Mathematics, University College Dublin, Ireland^cDépartement de Psychologie, Université de Montréal, Canada

ARTICLE INFO

Article history:

Received 23 November 2010

Received in revised form 6 April 2011

Available online 16 April 2011

Keywords:

Classification images

Vision

Facial expression categorization

Bubbles

Adaptive sampling

Reinforcement learning

Temporal difference learning

ABSTRACT

Bubbles is a classification image technique that randomly samples visual information from input stimuli to derive the diagnostic features that observers use in visual categorization tasks. To reach statistical significance, Bubbles performs an exhaustive and repetitive search in the stimulus space. To reduce the search trials, we developed an adaptive method that uses reinforcement learning techniques to optimize sampling by exploiting the observer's history of categorization. We compared the performance of the original and the adaptive Bubbles algorithms in a model observer and eight human adults who all resolved the same visual categorization task (i.e., five facial expressions of emotion). We demonstrate the feasibility of a substantial reduction (by a factor of ~ 2) in the number of search trials required to locate the same diagnostic features with the adaptive method, but only when the observer reaches a performance threshold of 50% correct for each expression category. When this threshold is not reached, both the original and adaptive algorithms converge in the same number of trials.

© 2011 Elsevier Ltd. All rights reserved.

1. Introduction

Classification image techniques are widely used in vision research to reveal the visual information subtending a range of tasks (Eckstein & Ahumada, 2001). One influential approach uses additive white noise to partially mask the original stimuli. Classification images are then obtained by summing over the noise fields weighted by the observer's responses (Abbey & Eckstein, 2002). Bubbles (Gosselin & Schyns, 2001) is a related technique in which Gaussian apertures are randomly located to reveal parts of a stimulus, a form of multiplicative noise. Since its introduction with faces (Gosselin & Schyns, 2001), Bubbles has been applied in various domains, including the categorization of natural scenes (McCotter, Gosselin, Sowden, & Schyns, 2005), the perceptual switching of ambiguous images (Bonnar, Gosselin, & Schyns, 2002), and several neuropsychological, single patient studies (Adolphs, Gosselin, Buchanan, Tranel, & Schyns, 2005; Caldara et al., 2005). In addition to behavioural studies, Bubbles has lately been applied to brain imaging signals in EEG (Schyns, Jentzsch, Johnson, Schweinberger, & Gosselin, 2003; Smith, Gosselin, & Schyns, 2006; Schyns, Petro, & Smith, 2007; van Rijsbergen & Schyns, 2009), fMRI (Smith et al., 2008) and MEG studies (Smith, Fries, Gosselin, & Schyns, 2009).

One general drawback of classification image techniques is the exhaustive search of the stimulus space (typically performed over several thousand trials) that is required to isolate the subspaces significantly correlated with performance. High trial numbers limit the range of applications, for example with clinical or children populations, or with brain imaging technologies, either because it is difficult to maintain the constant attention of young observers, or because it is too impractical to run observers across many sessions in clinical settings.

In simple discrimination or detection tasks, when the stimuli are of a small size, sophisticated estimators such as generalized linear models can be used to derive classification images from experimental data more efficiently and in fewer trials (Knoblauch & Maloney, 2008; Mineault, Barthelmé, & Pack, 2009). Here, to circumvent the practical problems of exhaustive Bubbles searches in complex recognition tasks, we developed an adaptive search algorithm based on reinforcement learning (RL, Sutton & Barto, 1998). This algorithm increases bubbles sampling efficiency by exploiting the observer's past performance with sampled information, i.e., the history of sampling. A MATLAB implementation of the algorithm is available from the internet (<http://www.psy.gla.ac.uk/hongfang/ABubbles/ABubbles.zip>).

2. Bubbles

The sampling process of the original Bubbles algorithm (Gosselin & Schyns, 2001) starts by generating a mask comprising a number of randomly located Gaussian apertures (the bubbles). A

* Corresponding author.

E-mail addresses: hongfang.wang@glasgow.ac.uk (H.F. Wang), nial.friel@ucd.ie (N. Friel), frederic.gosselin@umontreal.ca (F. Gosselin), p.schyns@psy.gla.ac.uk (P.G. Schyns).

piecewise multiplication of the mask with the original image (e.g., of a face) presents sampled stimulus information to the observer (as illustrated in Fig. 1). The observer then attempts to categorize the stimulus. This process is repeated over several thousand trials. By recording the observer's response to each sampled stimulus, credits are assigned to pixels revealed by those bubbles associated with correct categorization responses. Over the course of the experiment, the number of Gaussian apertures (i.e., the sampling density) is adjusted on each trial to maintain categorization accuracy at 75%. Following the experiment, bubble masks corresponding to correct responses are summed together to form a correct plane. All the masks generated for the experiment are also summed up to form a total plane. An elementwise division of the correct plane by the total plane assigns a probability value to each pixel of the input space (Gosselin & Schyns, 2001). This resulting proportion plane forms the classification image and a P -value statistics ascribes significance to the pixels diagnostic for the task (Chauvin, Worsley, Schyns, Arguin, & Gosselin, 2005). To derive a robust classification image, an exhaustive search of the information space is required with sufficient repetitions to derive stable proportions in the proportion plane. Consequently, with a typically sized image space of 256×256 pixels, usually no fewer than 2000 trials are required per observer to reach statistical significance in the sort of multiple alternative forced choice task that characterizes higher-level visual categorizations.

3. Reinforcement learning and adaptive bubbles

To progress from an exhaustive to an adaptive Bubbles search, our objective is to generate bubble masks in an adaptive manner. Practically, this entails learning to assign bubbles to the locations of the visual input space that lead to correct categorizations, i.e., those regions likely to represent *diagnostic information* (Schyns, Goldstone, & Thibaut, 1998).

To this end, the adaptive algorithm construes Bubbles as a variety of reinforcement learning techniques (Sutton & Barto, 1998). Reinforcement learning covers a class of problems in computational learning that are guided by a reward system or a goal. Such problems usually involve an agent and its environment. In general, the environment is represented as a state function, and it is also responsible for providing feedback (in terms of reward) to the agent. The role of the agent is to take actions on the basis of the state of the environment, the feedback it has received, and the policy currently in use.

In the context of Bubbles, the agent is the bubbles generator, which takes actions that probabilistically distribute bubbles in areas of the input space that maximize reward. The observer rewards (vs. punishes) the actions of the bubbles generator by responding correctly (vs. incorrectly) to the information sampled with these bubbles. The bubbles generator then updates the expected total reward associated with each bubble location and then generates the next distribution of bubbles for subsequent trials.

Specifically, we define the state space \mathcal{S} as the extension of the two-dimensional lattice of possible bubble locations: a space of 25×16 locations centred on a cell of 15×15 pixels of the original 375×240 pixels image (see Fig. 1b). The state space \mathcal{S} therefore comprises all possible configurations of a number of bubbles encountered over each trial of the experiment. The algorithm comprises two main modes of operation. In the *exploratory* mode, the bubbles generator randomly distributes Gaussian apertures (of standard deviation $\sigma = 16$ pixels) over the lattice, building from Smith, Cottrell, Gosselin, and Schyns (2005). Together with the observer's categorization responses, the exploratory mode derives an initial estimate of the distribution of diagnostic locations over the lattice. In the *exploitation* mode, the bubble generator exploits the accrued knowledge, and moves the bubbles on the lattice (restricting movements to local neighborhoods) to search for locations that optimize reward. Throughout learning, the algorithm alternates between exploration and exploitation, iterated in a number of learning episodes. The balance between exploration and exploitation is set by the agent's policy.

More formally, on each trial t , the bubble generator analyzes the bubbles in the lattice (which are in state $s_t \in \mathcal{S}$), consults its policy π to determine its operating mode (i.e., exploration or exploitation) and performs action a_t (resulting in specific moves of bubbles on the lattice, from the corresponding action space \mathcal{A}). Following the action, the bubbles generator receives from the environment a reward signal $r_{t+1} \in \mathbb{R}$, i.e., the categorization response of the observer, and moves to a new bubble configuration state $s_{t+1} \in \mathcal{S}$.

The learning process is guided by the goal to maximize the expected reward for each lattice location. For each state s , the value function represents the expected total reward that the agent would gain if it were to start from this state in the next trial. On each trial t , the agent updates the value function of the current state $V(s_t)$ of the lattice. The state values thus represent the probabilities of each pixel being diagnostic for the current categorization task, and so are regarded as the pixels of our "classification image".

The learning algorithm uses the temporal difference learning method ($TD(\lambda)$) (Sutton, 1988; Sutton & Barto, 1998) to update

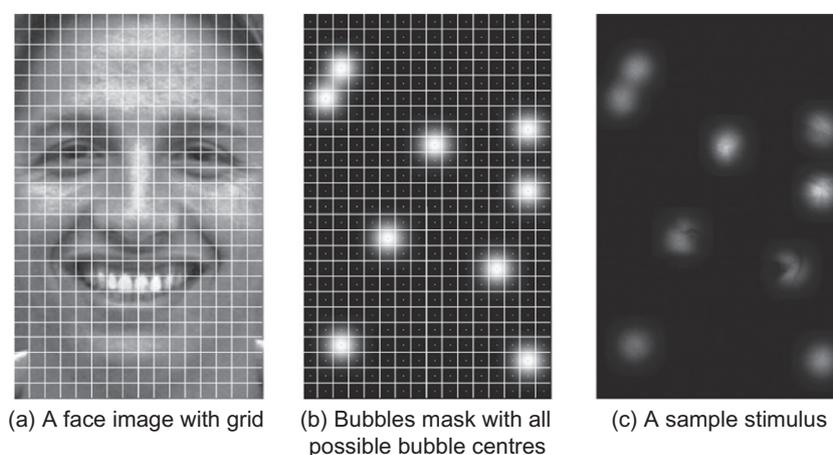


Fig. 1. (a) Original 375×240 pixels expressive face image with overlaid lattice of 15×15 pixels per cell. (b) Gaussian bubbles (of $\sigma = 16$ pixels) randomly distributed over the lattice, with each bubble centered on a lattice cell. (c) Mask of random bubbles sampling facial information from the original face image.

the state values $V(s)$. Here, $\lambda \in [0, 1]$ is a parameter that controls the strength of the updating feedback. When $\lambda = 0$ the learning process only uses the immediately preceding feedback for updating the state of the lattice: $TD(0)$ is

$$V(s_t) \leftarrow V(s_t) + \alpha(r_{t+1} + \gamma_t V(s_{t+1}) - V(s_t)), \quad (1)$$

where α is the learning rate, and $\gamma_t \in [0, 1]$ weighs the future reward into the estimate of the current lattice state. The future reward is approximated by the value of the next state. For each lattice location (and corresponding 15×15 image pixels), the current reward value corresponds to the Gaussian value of the sampling bubble that covers the lattice location and multiplied by a scalar of +1 (vs. -1) when the categorization is correct (vs. incorrect).

To balance exploration and exploitation during learning, we divided the process into a number of learning episodes – here, each episode comprised 75 trials randomizing 15 trials per expression. In each episode, the policy π decides between the exploratory and the exploitation mode. To this end, we first compute the diagnosticity ($D(s)$) of each bubbles location s as follows:

$$D(s) = \exp\left(\frac{1}{\beta} \sum_{i \in I_s} V(i)\right) \quad (2)$$

where β is a constant (set to 625 in our experiments) which controls how fast the learning process goes into the exploitation mode. $V(i)$ is the current estimate of state value of the pixels i within lattice location s . When either $std(V)$ or $D(s)$ is below a predefined threshold the exploratory mode distributed bubbles over the lattice according to a uniform distribution. When the state value and its standard deviation are both above thresholds, a softmax procedure implements the exploitation mode. Formally, the probability that the bubbles generator moves to state s'_t given that its current state is s_t is:

$$p(s'_t | s_t) = \frac{\exp\{D(s'_t)/\tau\}}{\sum_{s'' \in N_s} \exp\{D(s'')/\tau\}} \quad (3)$$

where $\tau = \frac{\theta}{k}$, k represents the number of repetitions of a given expression in the current episode, and θ is a constant larger than k . N_s is the neighborhood set of all possible states of state s , which is a 3×3 neighborhood window based on the lattice, centred on state s . At the beginning of the experiment, the bubble generator will use exploration and move bubbles randomly in the lattice within each episode. Fig. 2, Panel a, illustrates such random exploration. With accrual of diagnosticity, based on the observer's responses to sampled information, the generator will gradually move bubbles to

diagnostic lattice locations. Fig. 2 illustrates the situation when one of the bubbles moves progressively to the lattice location corresponding to the diagnostic eyes of “surprise” (Panel b) and to the mouth in “happy” (Panel c). Since a bubble moves probabilistically, it is likely that it will visit a non-diagnostic region after visiting a diagnostic region, as shown in Panel c of Fig. 2.

The learning process described above starts with a number of independent bubbles randomly located in the lattice. After each learning episode, this number is increased or decreased according to the observer's performance to maintain categorization accuracy at criterion (set to 75% correct in our experiments). Sampling bubbles are randomly relocated at the beginning of each learning episode. The learning rate α in Eq. (1) decays exponentially with the accumulation of diagnosticity as computed in Eq. (2). Upon starting exploitation, the algorithm records the number of exploitation episodes. When this number reaches a predefined threshold (6 in our examples with human observers), the learning task is considered as completed and the bubbles will be moved randomly in the lattice again. This is to ensure that the convergence of one expression will not provide extra cues to the observer about the other categorization responses and thus bias the results. The experiment stopped when the diagnostic features were learned for each categorization task, or when all predefined number of episodes were completed.

4. Experiments

4.1. Facial expression categorization with a model observer

To benchmark the adaptive algorithm, we built a model observer similar to that of Smith et al. (2005) and compared its performance to the non-adaptive original algorithm in a facial expression categorization task (*happy, surprised, disgusted, angry, and neutral*). We used a database of 50 original face images (5 male and 5 female, FACS coded (Dailey, Cottrell, & Reilly, 2001), with image size 375×240 pixels). We apposed a 25×16 lattice of possible bubble locations onto the original image space, where each lattice location represents a cell of 15×15 pixels (Fig. 1b). For both the adaptive and non-adaptive algorithms, we set the initial number of bubbles (with standard deviation = 16 pixels) to 10 for each expression. To maintain performance at a 75% categorization criterion, for each expression category we adjusted the number of bubbles after each episode.

For the learning rate α , in general a small number of is preferred and this number should decrease gradually as learning progresses

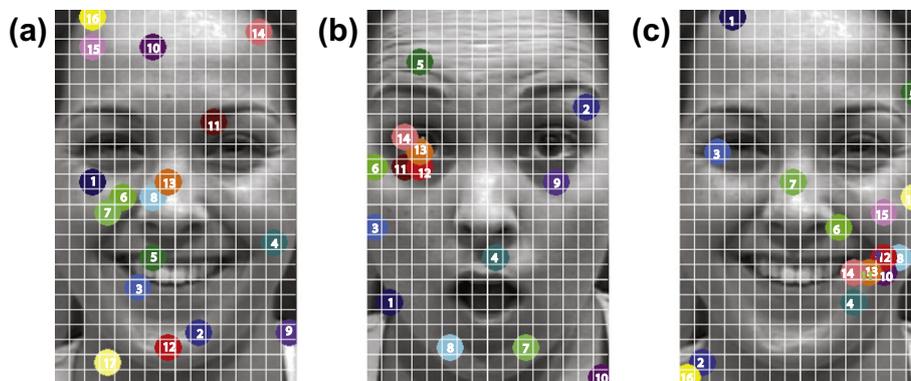


Fig. 2. Examples of how a particular bubble changes lattice positions over time course within one experimental episode. Numbers and colors (1–17 (a and c) and 1–14 (b)) indicate, for the expression and bubble considered, the movements of the bubble over the lattice, at each time step of a given learning episode. (a) *Exploration mode*. When the policy is in its exploration mode, the bubble moves randomly over the lattice. (b) *Exploitation mode*. When the policy has moved to its exploitation mode, the bubble progressively moves towards the diagnostic eye in “surprise”. (c) *Exploitation mode*. This illustrates that in some cases, exploitation can also be followed by exploration – i.e., a movement away from the diagnostic corner of the mouth in happy.

for convergence (see, for example, Sutton, 1988). However, the learning process might be too long if the learning rate is too small. A large value of learning rate might fail to converge. To illustrate the effect, we have run the simulation with α values initialized to 0.3, 0.198, 0.10, and 0.08, respectively, and decayed exponentially as the computed diagnosticity grew.

We added a fixed quantity of Gaussian white noise to a bubbled face as input to the model observer. The model observer then compared (i.e., cross-correlated) the input with all 50 possible original face images, themselves masked with the same bubbles as the input. The categorization response corresponded to the category of the face image with the highest correlation coefficient. The computer programs for the experiments were written with the Psychophysics Toolbox (Brainard, 1997; Pelli, 1997) in MATLAB.

4.2. Facial expression categorization with human observers

We performed a similar experiment with human observers. Observers categorized the input bubbled face by pressing a key labeled as *happy*, *surprised*, *disgusted*, *angry*, or *neutral*. Each stimulus was displayed in the centre of a computer screen with a light gray background until response. Ten paid adult subjects (four males), from the University of Glasgow with normal or corrected to normal vision participated in the experiments with both the adaptive and non-adaptive algorithms. All observers initially learned to categorize the original 50 face images on the same computer screen over 100 trials to confirm that they understood the task, that they could reach a 90% correct categorization rate, and became familiar with the response keys. Two observers (one male) could not reach a 90% correct categorization criterion and so were removed from the experiment.

A chinrest maintained a constant viewing distance of 0.7 m and viewing angle of $3.90^\circ \times 6.10^\circ$. We instructed observers to classify the stimuli as accurately as they could and to guess if they were unsure. Observers could take a break after every 75-trials episode and the experiment was divided into several sessions over a few days.

We counterbalanced the order of exposure to the adaptive and non-adaptive algorithms as follows: EB, LF, ASR and KW started with the adaptive algorithm; MBi, MB, TR and MC started with the non-adaptive algorithm. We set the initial number of bubbles to 9 for each expression and algorithm. This number drifted during the experiment (between 7 and 15) to maintain categorization at 75% correct for each expression. The learning rate α is set to 0.198. All other parameters of the algorithm and experiment were identical to those applied to the model observer.

4.3. Results

4.3.1. Classification images

We analyzed the results from the non-adaptive Bubbles algorithms as is typical: for each expression, we summed the masks of bubbles leading to correct responses and subtracted from this the sum of the masks of bubbles leading to incorrect responses. This leads to a distribution of difference scores across the face plane. For the adaptive algorithm, the value plane (i.e., the tabular form of the learned state value function for each categorization task) was the result plane. We then smoothed these result planes with a Gaussian kernel ($\sigma = 10$), z-scored the outcomes, applied a threshold with pixel test and $p \leq 0.05$ (Chauvin et al., 2005). We used the MATLAB toolbox *Stat4Ci* for the computations. Only clusters of at least 81 contiguous pixels were kept to minimize noisy clusters. The thresholded pixels represent the classification image solution of each algorithm (see Figs. 3, 4).

To compare the performance between the original and adaptive algorithms, we proceeded in two steps. First, we defined the *target*

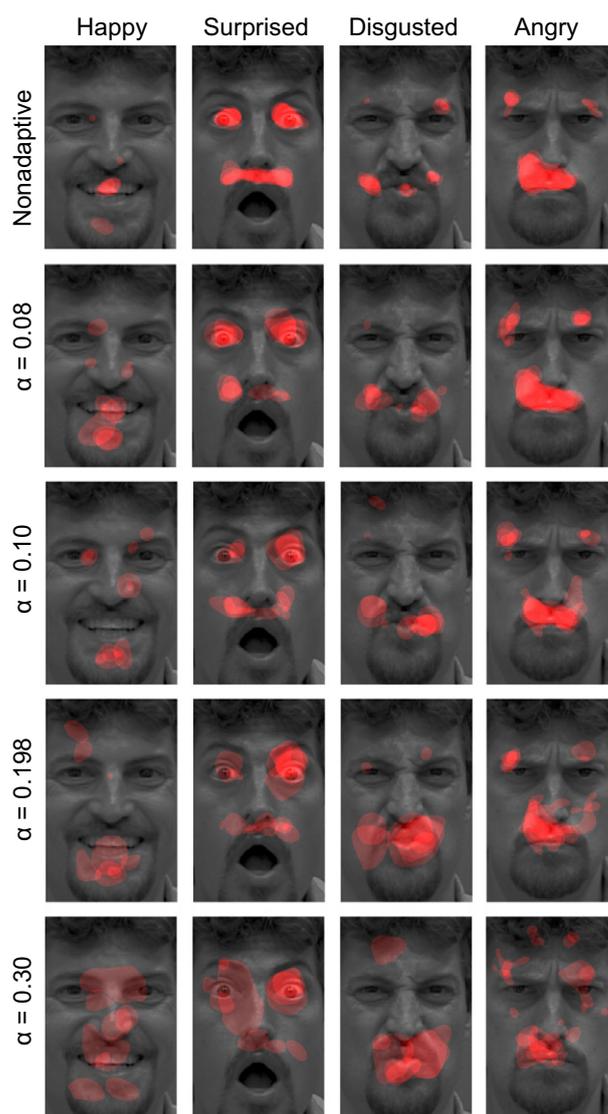


Fig. 3. Results from the model observer. Diagnostic features are highlighted in red. Colour intensity represents percentages of the corresponding pixel appearing in the five independent runs. From top to bottom: Diagnostic features from non-adaptive bubbles obtained after 37,500 trials; Diagnostic features obtained from adaptive bubbles when stopping criteria were met and when the learning rate α is set to $\alpha = 0.08$, 0.10, 0.198 and 0.30, respectively.

classification image for a given categorization (e.g. “happy”) as the classification image that is obtained when the stopping criteria are met (in the adaptive algorithm) and as the image obtained when all trials as completed in the original algorithm. Second, we derived the minimum number of trials required to reach a classification image that is sufficiently similar (to be explained below) to the expected target classification image (henceforth, we will refer to these trials as the *essential number of trials*). To derive the essential number of trials for each expression we computed the correlation coefficients ρ between the target classification images and the intermediate classification images obtained after each episode. A criterion threshold of $\rho \geq 0.75$ was used to determine sufficient similarity between an intermediate and a target classification image. The corresponding number of trials required to reach an intermediate classification image with $\rho \geq 0.75$, when all subsequent intermediate classification images still satisfy the criterion, corresponds to the *essential number of trials* for this category.

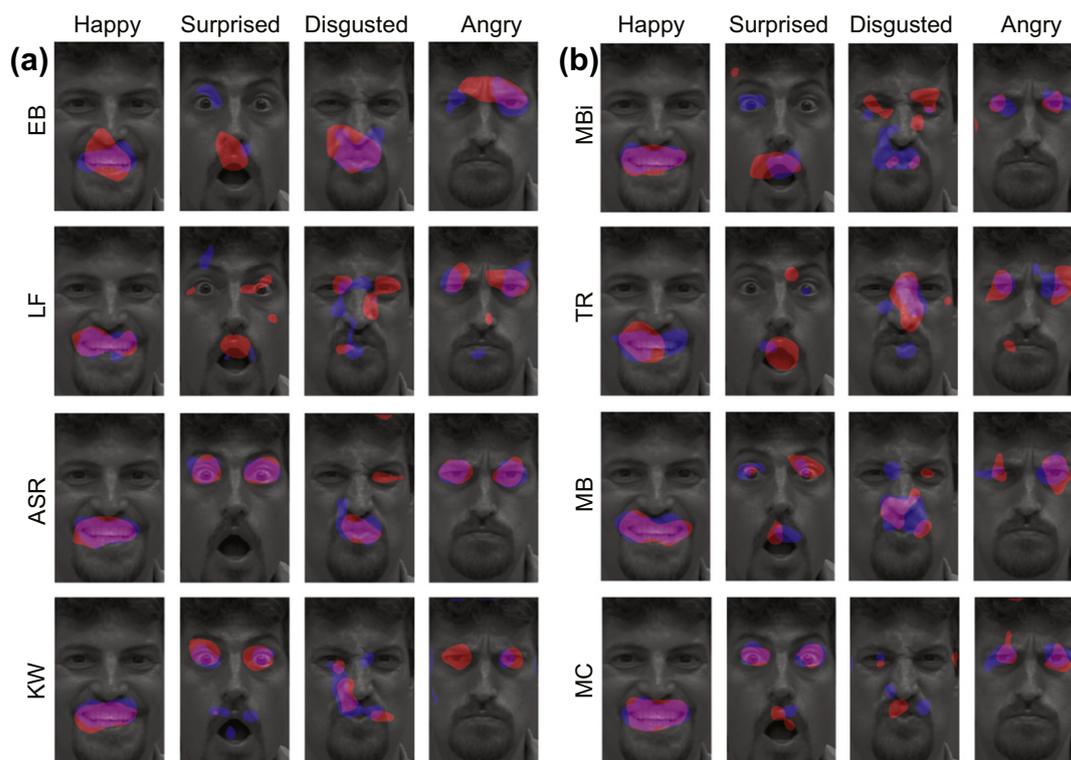


Fig. 4. Diagnostic features from classification images of human observers from adaptive bubbles (shown in red) and non-adaptive bubbles after 4500 trials (shown in blue), with overlapped features in magenta. (a) From top row to bottom row: observers EB, LF, ASR, and KW. (b) From top row to bottom row: observers MBi, TR, MB, and MC.

4.3.2. Model observer – essential number of trials

We first ran the model observer with the adaptive algorithm with a learning rate $\alpha = 0.198$ until the stopping criteria were met. We further ran the model observer for $\alpha = 0.3, 0.10$, and 0.08 , respectively, to examine the effect of the learning rate. For each α value, we repeated the experiment five times. As a comparison, we then run the model observer with the non-adaptive algorithm for 37,500 trials, and this experiment was again repeated five times. In all the above experiments, the MATLAB random number generator was set to a different state at the beginning of each run.

In Fig. 3, each image shows all the diagnostic features (highlighted in red) for the corresponding facial expression appearing in the five target classification images obtained from the five runs of each experiment condition. Colour intensity represents the frequency of the corresponding feature being detected in the five runs. The first row of the figure shows the features for facial expressions *Happy*, *Surprised*, *Disgusted*, and *Angry* obtained from the non-adaptive Bubbles. The remaining four rows show the results for each facial expression obtained from the adaptive algorithm with each α value when the stopping criteria were met.

In the above experiments, the average number of essential trials is 20,190 for the non-adaptive algorithm (25,545, 10,830, 30,045, and 14,340) for expressions *happy*, *surprised*, *disgusted*, and *angry*, respectively. The diagnostic features most similar to those from the non-adaptive experiment were obtained when the learning rate $\alpha = 0.08$ (see Fig. 3). The average numbers of essential trials from this case were 7395, 7440, 8835, and 8145, with an overall average trial numbers of 7953.75. That is, when the learning rate α is 0.80, the adaptive algorithm required only 39.39% of the non-adaptive trial numbers to reach similar diagnostic features for the same categorization task, i.e., it is about 2.54 times faster. As the learning rate increases, the number of essential trials decreases. However, the diagnostic features obtained from a larger

learning rate are not as accurate as those obtained from a smaller learning rate, or those obtained from the non-adaptive algorithm (see Fig. 3).

4.3.3. Human observers – essential number of trials

To compare performance between the two algorithms with human observers, we computed classification images in one of two possible ways. First, the adaptive algorithm stopped early when the considered expression had a correct rate above 60% (see Table 1, the number of trials associated with each observer and expression). Second, when the adaptive algorithm did not reach stopping criterion (see crosses in Table 1), we computed the diagnostic features after the 3000 trials limit. Visual inspection of Fig. 4 reveals

Table 1

Number of trials for each observer to reach the stopping criterion for a target classification image for each expression with the adaptive bubbles algorithm. (For interpretation of color in Table 1, the reader is referred to the web version of this article.)

	Happy	Surprise	Disgust	Angry
Model	7395	7440	8835	8145
EB	1125	1350	1500	2100
LF	2025	1425	2250	3000
TR	1575	1275	1800	1875
MB	1950	1575	2175	×
MBi	1800	1875	×	×
ASR	1875	2700	×	×
KW	1950	2550	×	×
MC	2475	×	×	×

Note: The numbers are the total number of trials which include all five categories. Cells with a '×' entry means the stopping criterion was not met. In the model observer case, the results are the averages when $\alpha = 0.08$. Font color encodes the corresponding correct categorization rate: <40%, 40–50%, 50–60%, and 60–85%.

Table 2

Essential number of trials for three observers to reach a classification image for each expression with both algorithms. (For interpretation of color in Table 2, the reader is referred to the web version of this article.)

	Happy	Surprise	Disgust	Angry
EB/1	900	1350	1200	2025
EB/2	2025	4125	2025	3000
LF/1	1500	1350	1875	1425
LF/2	1575	3450	4125	2175
TR/1	1350	1200	1650	1725
TR/2	1500	4425	3450	3075

Note: The trial numbers in the table are the total number of trials. xx/1 means subject xx with adaptive bubbles, and xx/2, non-adaptive bubbles. Font color encodes the corresponding correct categorization rate; <40%, 40–50%, 50–60%, 60–85%, and >85%.

that the two algorithms converged on nearly identical features, ensuring that the adaptive algorithm did not adversely bias observers.

When the adaptive algorithm reached its stopping criterion for all expressions (i.e., for observers EB, LF and TR), we computed the essential number of trials using the same criteria as in the model. As Table 2 illustrates, on average, the adaptive algorithm required only 50.21% of trials compared with the non-adaptive version.

5. Summary

In this paper we have developed and tested a new adaptive sampling method that learns to sample bubbles judiciously, based on the observer's history of categorization behaviour. The algorithm reduces the number of trials required to reach a diagnostic image by finishing the experiment as soon as it has learned the categorization information. The algorithm is framed in the reinforcement learning family and is model free. This guarantees its results are free from experimenter bias.

We tested the algorithm with five classic facial expression categorizations with a model observer and eight human observers. The outcome revealed a lower number of trials in the experiments (i.e., 39.39%) with the model observer and four different learning rates with similar diagnostic information extracted. For human observers, the number of trials was also reduced (with $\alpha = 0.198$), especially so for observers who reliably categorized the expressions, with few confusions among different facial expression categories. Three such observers had a correct categorization rate above 50% for each expression and only required an average of 50.21% (i.e., 293 trials per observer per expression) of the non-adaptive number of trials to reach similar diagnostic features. Pooling across observers, this faster rate of convergence accounted for 68.75% (22 out of 32) of the total expressions tested. The number of experimental trials required with our adaptive algorithm is thus much less than that in any of the original Bubbles experiments for which no less than 1000 trials (and sometimes even 3000) per expression per observer, e.g., (Gosselin & Schyns, 2001; Smith et al., 2005; Schyns et al., 2007) were needed.

For observers with very low correct categorization rates (e.g., <40%), the resulting classification images are similar to those obtained with at least the same number of trials and non-adaptive Bubbles.

In sum, our adaptive algorithm has achieved its target by reducing the number of required trials by a factor of about 2, but the exact reduction in trial number depends on observer's categorization accuracy.

References

- Abbey, C. K., & Eckstein, M. P. (2002). Classification image analysis: Estimation and statistical inference for two alternative forced choice experiments. *Journal of Vision*, 2(1), i.
- Adolphs, R., Gosselin, F., Buchanan, T. W., Tranel, D., & Schyns, P. G. (2005). A mechanism for impaired fear recognition after amygdala damage. *Nature*, 433, 68–72.
- Bonnar, L., Gosselin, F., & Schyns, P. G. (2002). Understanding Dali's slave market with the disappearing bust of voltaire: A case study in the scale information driving perception. *Perception*, 31, 683–691.
- Brainard, D. H. (1997). The psychophysics toolbox. *Spatial Vision*, 10, 433–466.
- Caldara, R., Schyns, P. G., Mayer, E., Smith, M. L., Gosselin, F., & Rossion, B. (2005). Does prosopagnosia take the eyes out of face representations? Evidence for a defect in representing diagnostic facial information following brain damage. *Journal of Cognitive Neuroscience*, 17(1), 1652–1666.
- Chauvin, A., Worsley, K. J., Schyns, P. G., Arguin, M., & Gosselin, F. (2005). Accurate statistical tests for smooth classification images. *Journal of Vision*, 5(9), 659–667.
- Dailey, M., Cottrell, G. W., & Reilly, J. (2001). CALifornia Facial Expressions (CAFE). <<http://www.cs.ucsd.edu/users/gary/CAFE/>>
- Eckstein, M. P., & Ahumada, A. J. (2001). Classification images: A tool to analyze visual strategies. *Journal of Vision*, 2(1), i.
- Gosselin, F., & Schyns, P. G. (2001). Bubbles: A technique to reveal the use of information in recognition tasks. *Vision Research*, 41, 2261–2271.
- Knoblauch, K., & Maloney, L. T. (2008). Estimating classification images with generalized linear and additive models. *Journal of Vision*, 8(16), 1–19.
- McCotter, M., Gosselin, F., Sowden, P., & Schyns, P. G. (2005). The use of visual information in natural scenes. *Vision Cognition*, 12(6), 938–953.
- Mineault, P. J., Barthelmé, S., & Pack, C. C. (2009). Improved classification images with sparse priors in a smooth basis. *Journal of Vision*, 9(10), 1–24.
- Pelli, D. G. (1997). The VideoToolbox software for visual psychophysics: Transforming numbers into movies. *Spatial Vision*, 10(4), 437–442.
- Schyns, P. G., Goldstone, R. L., & Thibaut, J. P. (1998). The development of features in object concepts. *Behavioral and Brain Science*, 21(1), 1–17.
- Schyns, P. G., Jentzsch, I., Johnson, I., Schweinberger, S. R., & Gosselin, F. (2003). A principled method for determining the functionality of ERP components. *Neuroreport*, 14, 1665–1669.
- Schyns, P. G., Petro, L. S., & Smith, M. L. (2007). Dynamics of visual information integration in the brain for categorizing facial expressions. *Current Biology*, 17(18), 1580–1585.
- Smith, M. L., Cottrell, G. W., Gosselin, F., & Schyns, P. G. (2005). Transmitting and decoding facial expressions. *Psychological Science*, 16(3), 184–189.
- Smith, M. L., Fries, P., Gosselin, F., & Schyns, R. G. P. G. (2009). Inverse mapping the neuronal substrates of face categorizations. *Cerebral Cortex*, 19(10), 2428–2438.
- Smith, M. L., Gosselin, F., & Schyns, P. G. (2006). Perceptual moments of conscious visual experience inferred from oscillatory brain activity. *PNAS*, 103(14).
- Smith, F. W., Muckli, L., Brennan, D., Pernet, C., Smith, M. L., Belin, P., et al. (2008). Classification images reveal the information sensitivity of brain voxels in fMRI. *NeuroImage*, 40(4), 1643–1654.
- Sutton, R. S. (1988). Learning to predict by the methods of temporal differences. *Machine Learning*, 3(9), 9–44.
- Sutton, R. S., & Barto, A. G. (1998). *Reinforcement learning: An introduction*. Cambridge: The MIT Press.
- van Rijsbergen, N., & Schyns, P. G. (2009). Dynamics of trimming the content of face representations for categorization in the brain. *PLoS Computational Biology*, 5(11), e1000561.