I-POMDP: An Infomax Model of Eye Movement

Nicholas J. Butko

Department of Cognitive Science
University of California, San Diego
La Jolla, CA 92093-0515
Email: nbutko@cogsci.ucsd.edu

Javier R. Movellan

Institute for Neural Computation
La Jolla, CA 92093-0445
Email: movellan@mplab.ucsd.edu

Abstract—Modeling eye-movements during search is important for building intelligent robotic vision systems, and for understanding how humans select relevant information and structure behavior in real time. Previous models of visual search (VS) rely on the idea of "saliency maps" which indicate likely locations for targets of interest. In these models the eyes move to locations with maximum saliency. This approach has several drawbacks: (1) It assumes that oculomotor control is a greedy process, i.e., every eye movement is planned as if no further eye movements would be possible after it. (2) It does not account for temporal dynamics and how information is integrated as over time. (3) It does not provide a formal basis to understand how optimal search should vary as a function of the operating characteristics of the visual system. To address these limitations, we reformulate the problem of VS as an Information-gathering Partially Observable Markov Decision Process (I-POMDP). We find that the optimal control law depends heavily on the Foveal-Peripheral Operating Characteristic (FPOC) of the visual system.

I. Introduction

Personal robots must juggle the varied demands of everyday life in an intelligent way. To do this effectively, they must continuously make choices about which sensory data can be ignored and which data must be processed. Camera movements are a prototypical attentional control mechanism, selecting small regions of the visual world to process moment-to-moment. Moreover, personal robots that move their sensors in a purposive manner tend to appear intelligent and lifelike to the humans interacting with them (Figure 1b).

Studying human eye-movement highlights principles that may be important for robots. Past years have seen a large growth of publications on computational models of visual search that rely on the idea of visual saliency. These models are evaluated in terms of how well they describe where humans tend to fixate within an image or video (Figure 1a). Recently, these models have become fast enough to run in low-end computers in real time and at little computational cost, while maintaining competitive accuracy [1]. This makes it possible to provide robots with principled mechanisms to choose where to look, and still have processing power left over for other tasks.

Models of visual salience can usually be categorized as *descriptive*, or *prescriptive*. Descriptive models try to match human data directly, either by following psychological theories (e.g. [2] which models Feature-Integration-Theory [3]), or by directly fitting models to human data [4].

(a) Example Saliency Map   (b) Robot Joint Attention

Fig. 1: (a) The saliency-map approach to robot eye-movements models where humans would look in an image. *Top*: The original image is the input to a saliency algorithm. *Bottom*: Bright regions are found salient by the algorithm. (b) When robots move their eyes in a fashion similar to humans, compelling feelings of intelligence are created, and joint attention is observed.

In contrast, prescriptive saliency models try to uncover the *underlying computational objectives* that organize oculomotor control. A popular choice is to frame the goal of eye-movement as Visual Search (VS), i.e. finding targets within a visual array. These models postulate that an intelligent agent should fixate its visual sensors on regions of the array where the target is most likely to be, given visual features across the whole array. Under this class of models saliency for each pixel x in the visual array is related to probability that it is rendered by a class of interest, $C_x = 1$. This can be framed mathematically as

$$\begin{aligned} \text{Salience}(x) &= p(C_x = 1 | Image) \\ &= \frac{p(Image | C_x = 1)p(C_x = 1)}{p(Image)} \end{aligned} \quad (1)$$

Many salience algorithms, e.g. [5]–[8], can be seen as special cases of this framework. See [1] for a more thorough review.

While the above framework is compelling and predicts well where on average humans will look in unconstrained tasks, it has important limitations: (1) The framework is atemporal, i.e. it gives no principled account for the order of saccades. (2) There is no mechanism for integrating information across

fixations. After a robot saccades, how does it update its saliency map based on what it saw? In fact, saliency models always fixate the same maximally salient location unless *Inhibition of Return* (IOR) is added. In practice, this means salience is subtracted around the currently fixated location. The use of IOR is an *ad hoc* procedure rather than an emerging property derived from a formal computational framework. (3) Existing salience models are not explicit in their assumptions of the Foveal-Peripheral Operating Characteristic (FPOC) of the visual system. As such they cannot account for how behavior should differ across systems with different FPOCs, like human infants, human adults, and robots. (4) Many existing saliency models assume what appears to be a reasonable control strategy: foveation of the most probable target locations. However it is possible that one can gain more information, in the long run, by foveating less probable target locations.

To address these limitations, we frame oculomotor control as a problem in stochastic optimal control. We start with a psychophysical model of visual perception proposed by Najemnik & Geisler [9] and reformulate it as an *Information-gathering Partially Observable Markov Decision Process* (I-POMDP). We design an I-POMDP with parameters fit to human data, as well as one designed to model robotic vision. We find that for both cases the optimal control laws avoid foveating locations where targets are likely to be. This suggests that postulates accepted by most of the current models of visual search may not be optimal. The proposed approach explains how looking behavior changes with the characteristics of the visual system, in particular the relative resolution of the foveal and peripheral regions. This argues against the "one-size-fits-all" approach taken previously that assumed that robotic eye-movement strategies should be the same as those in humans.

II. VISUAL SEARCH MODEL

Najemnik & Geisler [9] modeled VS as a control strategy designed to detect the location of a visual target under sensor uncertainty. This model assumes that there are N locations in the visual array where a visual target can appear. These locations have some spatial layout, e.g. on points of a grid. At every timestep, an agent can choose one of the points in the visual array to fixate. The agent will get information about whether or not the target is located near the fixation point with high certainty, and at increasingly distant points with decreasing certainty. An important limitation of their approach is that it assumes a greedy control strategy, i.e., each eye movement is planned under the assumption that no further eye movements will be possible.

Here we extend Najemik & Geisler's framework and reformulate it as a POMDP in which the primary goal is to gather information. We refer to this approach as I-POMDP. The new formulation allows us to answer questions about the moment-to-moment temporal dynamics of optimal eye-movement. A POMDP is defined by the following elements [10] (with their correspondences in the VS model):

- S: A set of states that cannot be directly observed by the agent ($S_t = i \in \{1 : N\}$ corresponds to the event that the target is at location i at time t).
- A: A set of actions that the agent can take ($A_t = k \in \{1 : N\}$ corresponds to the event that the agent's center of fixation is at location k at time t).
- O: A set of observations that can be made by the agent ($\vec{O}_t \in \mathcal{R}^N$ is a vector with elements O_t^j that correspond to noisy sensor evidence at time t about whether or not the target is at location j in the visual array).
- $p(S_{t+1}|S_t, A_t)$: Dynamics – How the state changes based on the agent's actions. (In the current task the visual target does not move, so $p(S_{t+1}|S_t, A_t) = 1$ if $S_{t+1} = S_t$, 0 otherwise. I.e., the state transition matrix is the identity matrix).
- $p(O_t|S_t, A_t)$: Observation model – How states and actions combine to yield observations (Section II-A).

A critical concept in POMDPs is the "Belief State" vector $\vec{B}_t \in [0,1]^N$, in which the ith element B_t^i represents the probability that the target is in visual array location i given all of the agent's previous eye movements and observations, i.e., $B_t^i \stackrel{\text{def}}{=} p(S_t = i|A_{1:t-1}, O_{1:t-1})$. A well known result in the theory of POMDPs specifies that the belief state vector at time t can be calculated based only on a single previous eye movement A_{t-1}, a single observation of the visual array $\vec{O}_{t-1}$, and the previous belief vector $\vec{B}_{t-1}$. Specifically,

$$B_t^i \propto p(\vec{O}_t|S_t = i, A_{t-1}) \sum_{j=1}^{N} p(S_t = i|A_{t-1}, S_{t-1} = j)B_{t-1}^j$$

In the current task the target never moves, and this becomes

$$B_t^i \quad \propto \quad p(\vec{O}_t|S_t = i, A_{t-1})B_{t-1}^i \qquad (2)$$

Equation (2) tells us that the belief state encodes all the relevant history of an agent's actions and observations. Thus by maintaining a representation the same size as a single observation, an agent can remember all that's relevant from all previous observations.

A. Observation Model

We formalize the operation of the visual array according to a signal-detection model adapted from [9]. A noisy observation $o_t^j \in \mathcal{R}$ is sampled at each potential target location j at each timestep t. In locations without a target, an observation is drawn from a "no activity" Gaussian distribution, which has zero-mean. Only the single observation directly at the target location is drawn from the "activity" Gaussian, which has a mean that increases as the target approaches the foveal region

$$p(o_t^j|S_t \neq j, A_t = k) = N(\mu = 0, \sigma^2 = 1) \qquad (3)$$
$$p(o_t^j|S_t = j, A_t = k) = N(\mu = d_{j,k}, \sigma^2 = 1) \qquad (4)$$

where $d_{j,k}$ is the Foveal-Peripheral Operating Characteristic (FPOC) of location j given that the retina is centered at k. This number indicates how easy it is to discriminate an observation rendered by the target according to (4) from an observation rendered by the background according to (3). In humans the
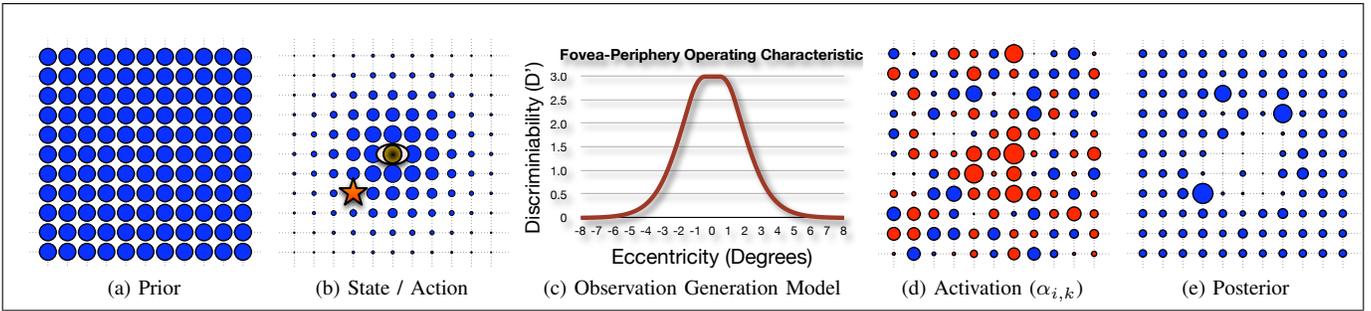
Fig. 2: The Visual Search POMDP: **(a)** Before the agent has looked at the visual array, it thinks the target could be at any location with equal likelihood. **(b)** At time t, the target (star) is located at $S_t = i$. Without knowing where the target is, the agent chooses a location $A_t = k$ to fixate (eye). This yields evidence about the location of the target with varying degrees of certainty $d_{j,k}$ at each location j (circles). **(c)** The eccentricity of the target from the center of fixation determines how easy it is to distinguish from background noise observed throughout the visual field. We use an FPOC fit by [9] to human subjects' data. **(d)** Activation levels $\alpha_{i,k}$ [Equation (5)] based on observations drawn according to the observation model [Equations (3)&(4)]. Blue circles (positive values) are more likely to be caused by a target; red circles (negative values) are more likely to be the result of only noise. Large red values near fixation provide strong evidence that the target is not where the agent is currently fixating. **(e)** The observation is combined with the Prior using Equation (6) to yield a new estimate of the likelihood of the target's location. This posterior exhibits IOR in that regions around the previous fixation have depressed probability if the object of interest is not seen there.

FPOC $d_{j,k}$ decreases with increased distance of location j from the current point of fixation k, meaning farther from the point of fixation it becomes harder to discriminate an observation caused by a target-based activity from one caused by noise alone. We used an FPOC function that was fit to human data by [9] (Figure 2c).

Under Geisler's model, the individual observations O_t^j are conditionally independent given the external scene,[1] and so

$$p(\vec{o}_t|S_t = i, A_t = k) = \prod_{j=1}^{N} p(o_t^j|S_t = i, A_t = k)$$

$$= 1/\sqrt{2\pi} \exp((o_t^i - d_{i,k})^2/2) \prod_{j \neq i} 1/\sqrt{2\pi} \exp((o_t^j)^2/2)$$

$$= \frac{\exp((o_t^i - d_{i,k})^2/2)}{\exp((o_t^i)^2/2)} Z$$

$$= \exp(\alpha_{i,k} d_{i,k}) Z; \quad \alpha_{i,k} \stackrel{\text{def}}{=} (o_t^i - d_{i,k}/2) \quad (5)$$

where Z is identical for all i, k. Combining this with Equation 2 yields the proportional belief update

$$B_t^i \quad \propto \quad \exp(\alpha_{i,k} d_{i,k}) B_{t-1}^i \quad (6)$$

The "activation value" $\alpha_{i,k}$ gives intuition about belief updates in the I-POMDP: if $\alpha_{i,k}$ for an element i of the observation vector is positive, the belief that the target is at location i will increase, while if it is negative, the agent will infer a reduced probability that the target is at location i.[2] (Figure 2d).

Note the simplicity of the belief update, Equation (6). Even though the I-POMDP has a large state, observation, and action space, updating beliefs is computationally quick. To calculate the relative probability that an entire observation vector was caused by a state, we need constant time (only a single element

---

of that observation vector is considered). Thus the process of computing the belief update for all all beliefs grows linearly.

III. LEARNING WHERE TO LOOK

We formulate VS as the process of gathering information about the location of the target of interest. According to the Infomax Principle [11], this is equivalent to minimizing the long term entropy of the belief-state $\vec{B}_t$. Thus, our goal is to learn a policy $\pi(\vec{B}_t) \rightarrow A_{t+1}$ that optimally decreases of $\vec{B}_t$ in at most T timesteps, where T is a planning horizon. I.e., it maximizes

$$V(\vec{B}_{t:T}) \quad = \quad \sum_{j=t}^{T} R(\vec{B}_j) \quad (7)$$

$$R(\vec{B}_t) \quad \stackrel{\text{def}}{=} \quad \sum_i B_t^i \log B_t^i \quad (8)$$

Algorithms for learning exactly optimal policies in POMDPs exist, but are only feasible with few states, actions, and observations [10]. Point-based methods can learn approximately optimal policies for POMDPs with many states and actions, but require few observations [12]. The I-POMDP model has an $\mathcal{R}^N$ observation space, which is very large. Moreover, these algorithms capitalize on the guarantee of traditional POMDPs that the reward function be linear in the belief vector $\vec{B}_t$; I-POMDPs allow non-belief-linear reward functions like the Infomax Reward function, Equation (8).

A. Policy Gradient

Due to the limitations of these approaches, here we consider function approximation methods which find locally optimal policy functions over a parameterized family of functions [13]. In particular we investigated Policy-Gradient methods which perform stochastic gradient ascent on the empirical

---

[1]Note this does not require that the observations are independent. The assumption would be satisfied, for example, if the sensors were noisy and the noise in each sensory element were independent.

[2]Before renormalization.

value function

$$\nabla_\theta \mathbf{E}[V(S)|\theta] = \nabla_\theta \sum_s V(s)p(s|\pi(s;\theta))$$

$$= \sum_s \left[\left(\sum_i r(s_i) \right) \left(\sum_i \frac{\nabla_\theta p(s_i|\pi(s;\theta))}{p(\vec{s}_i|\pi(s;\theta))} \right) p(s) \right] \quad (9)$$

An unbiased estimate of this gradient can be obtained by sampling state trajectories $\vec{s}$. For episodic problems, this results in a simple update procedure:

1) Set $t = 0$; Initialize X, the state of the MDP; Set $\vec{z} = \vec{0}$
2) For each state transition $X \to X'$ under policy $\pi(X;\theta)$
   - If X' is a final state or $t = T$, go to 1
   - Set $\vec{z} \leftarrow \vec{z} + \beta \frac{\nabla_\theta p(X'|X,\pi(s;\theta))}{p(X'|X,\pi(s;\theta))}$
   - Set $\vec{\theta} \leftarrow \vec{\theta} + \gamma_t r(X')\vec{z}$
   - Set $t \leftarrow t + 1$
   - Set $X \leftarrow X'$

where γ_t is a learning rate which can anneal over time, and β is a "bias-variance trade-off" parameter similar to λ in $TD(\lambda)$.

B. Policy Gradient with Logistic Policies

We chose to parameterize the policy as a logistic function:

$$p(A_{t+1} = i|\vec{B}_t;\theta) = \frac{\exp(\theta_i \cdot \vec{B}_t)}{\sum_{j=1}^n \exp(\theta_j \cdot B_t)} \quad (10)$$

This class of functions can be thought of as a neural network, with an input layer (the belief vector) projecting to an output layer in which each unit represents the probability of fixating a given location. The model is parameterized by θ, an $N \times N$ matrix, where N is the size of the visual array. Logistic policies extend many of the policies assumed in previous models (e.g., greedy search, random search) while allowing an intuitive examination of the learned policy. For example, a reasonable policy might be "look directly where the probability of the target is largest." We could verify whether this policy was optimal by examining the learned parameter matrix for very large values on the diagonal. This would mean that high belief that a target is at a given location leads to a high probability of fixating that location. Meanwhile connections to nodes at farther distances would taper off.

C. Shift & Rotation Invariance

Logistic policies can have many parameters. For an 11×11 visual array, there are $121 \times 121 = 14,641$ parameters. Figure 3 shows that it is indeed possible to learn a good policy in such a situation, but it takes a long time. The search space can be reduced to 61 parameters by exploiting the shift- and rotation-invariances of most visual search problems. This approach results in a convolutional policy which is defined by a rotationally symmetric two-dimensional kernel. Under convolutional policies of this type the value of a belief map is obtained by filtering with a filter whose impulse response equals the policy's two-dimensional kernel.

Gradients for a convolutional policy can be learned via weight-sharing, by tying the parameters of all connections to locations equidistant from the point of fixation. Learning
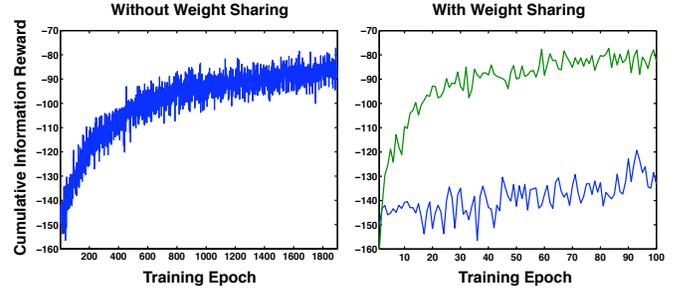

Fig. 3: Policy gradient enables learning even when there are 14,641 parameters. Learning is 20 times faster when we use weight sharing to exploit invariances, reducing the number of parameters to 61. The original learning curve is duplicated in blue in "With Weight Sharing" to highlight this timescale difference.

converges much faster (Figure 3). For the remainder of this paper, we use convolutional logistic policies learned by policy gradient with weight-sharing. We learn similar control laws regardless of initial parameters and visual array size, and so the approach seems robust to local minima in parameter space.

IV. EXPERIMENTS

To compute policies, we used a time-horizon T that was the same as the number of states N; the reward went to 0 long before T, approximating undiscounted infinite horizon. The parameter β was 0.75, γ was 0.02, and gradients were pooled across 150 episodes per epoch. We manipulated

- **Size of visual array:** The visual array size was 7×7 or 11×11, with $N = 49$ and $N = 121$ respectively.
- **Reward Function:** We compared the Infomax reward function with that postulated in Saliency literature.
- **Visual System Properties:** In addition to using an FPOC from psychophysical data [9], we studied what would happen in systems with different FPOCs.

Our results were analyzed in two ways.

- **Performance:** Performance was measured as "% Correct on an N-Alternative Forced Choice task (N-AFC)". That is, in an 11×11 visual array, if the location of the target had higher belief than all 120 other locations, the agent was right, otherwise it was wrong.
- **Control Law:** A policy is defined by a convolution kernel. If the kernel has a high value at eccentricity e, the agent wants to look toward some location k when there are high beliefs at locations e units away from k. If the kernel has a negative value at eccentricity e, the agent wants to look away from location k if there are high beliefs at locations e units away from k.

V. RESULTS

A. Performance & Policy

We first compared the constrained-optimization I-POMDP control law with two heuristic policies previously proposed.
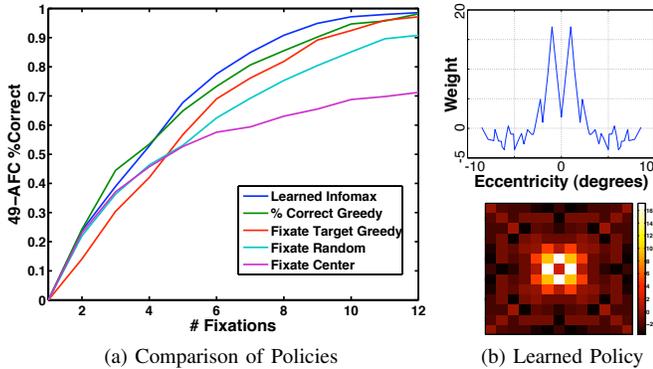
(a) Comparison of Policies   (b) Learned Policy

Fig. 4: **(a)** The Learned Policy performs better than 4 alternative policies described in Section V-A. Policy "% Correct Greedy", proposed in [9], outperforms the learned policy in only the first 4 fixations. This reflects the classic tradeoff between greedy and long-term planning. **(b)** The "receptive field" of the learned policy. *Top:* 1-D kernel function that was learned: The learned strategy looks *next to* places of high probability. *Bottom:* Rotating this kernel radially gives the radially symmetric 2-D convolution filter that defines the policy.

1) **Percent-Correct-Greedy:** Choose the action that yields the highest expected-percent-correct after the observation, i.e. that maximizes $R_{t+1} = \max_i E[B_{t+1}^i]$ (proposed in [9]). Computing a single action from this policy is $O(KN^3)$, where N is the size of the visual array and K is a very large constant. Because of the difficulty in computing this policy for each action, we used small 7×7 visual arrays.

2) **Fixate Target:** Choose the action k that maximizes the reward function $R_t = B_t^k$. This policy is implicit in visual saliency models like [1], [2].

We also evaluated the performance of two policies that we expected to perform poorly:

1) **Fixate Random Locations.**
2) **Fixate Center of Visual Array:** This policy discovers targets in the foveal region quickly, in the parafoveal region slowly, and in the peripheral region never.

The learned Infomax optimal controller reached high levels of accuracy (90% correct on the 49-AFC task) about 1.1 fixations earlier than the Percent-Correct-Greedy policy and about 3.5 fixations earlier than the Random policy (Table I). The performance of all policies is shown in Figure 4a.

TABLE I: # Fixations to reach 90% Correct (49-AFC)

Learned Infomax	% Cor. Greedy	Fixate Target	Fixate Random
7.86	8.96	9.25	11.33

The policy that achieves this high performance is visualized in Figure 4b. Interestingly, this policy chooses to foveate *next to* but *not at* locations where the target is likely to be. This appears to ensure that the target remains in the foveal region, while gathering extra information about the periphery.
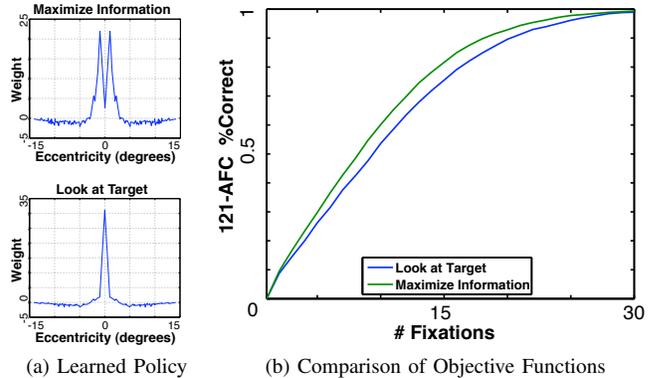

(a) Learned Policy   (b) Comparison of Objective Functions

Fig. 5: Performance loss from directly fixating the target; the visual array is 11×11. **(a)** Learned "receptive fields." *Top:* The Infomax policy closely resembles the policy in Figure 4b which was trained on a smaller visual array. *Bottom:* A different policy is learned when the goal is to look directly at the target. **(b)** Maximizing information performs noticeably better than trying to look directly at the target.

B. Comparison to Previous Approaches

The optimal control law avoids looking directly at the target, preferring to look just to the side of it. This contradicts a commonly accepted postulate that ideal searchers should directly fixate locations most likely to contain the search target. Such a strategy turns out to be suboptimal when more than one eye movement is possible. How much benefit does the ideal controller get by avoiding looking directly at the target?

When we evaluated the "Fixate Target" strategy previously, we did so in a greedy way after the fashion of the saliency literature. In order to be more fair to this strategy, we trained a controller that was given reward of 1 for looking directly at the target and 0 otherwise. Since the controller did not have direct access to the state, it received expected reward based on its belief state after the fashion of POMDPs [10], and so was linear in the belief state. This reward was the probability that it was looking at the target, $R_t = B_t^k$ where $A_t = k$.

We trained Infomax and Fixate Target controllers on an 11×11 visual array I-POMDP. The learned control laws are visualized in Figure 5a. The shape of the Infomax control law is similar to that of the 7×7 task, preferring to look next to the target. This indicates that the ideal strategy remains constant with problem size. The ideal Fixate Target strategy is different: when it is completely unsure of the target location, it has a slight preference to look in between candidate locations; otherwise it looks exactly where the target is likely to be. Figure 5b indicates that this is a reasonable but suboptimal strategy. Controllers optimized to Fixate Target require 20 fixations to reach 90% accuracy on a 121-AFC tasks, while those optimizing information-gain require 18 fixations.

This quantifies the expected performance boost achievable over previous Saliency approaches in robots [1], which attempted to look at search targets. Instead, our results suggest that a better strategy is to look *near but not at* visual targets. This presents avenues for psychophysical study, to see whether
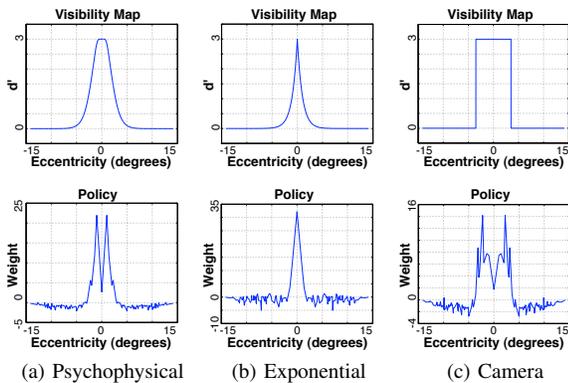
Fig. 6: Optimal policies (bottom) given different FPOCs (top). The visual array is 11×11. Each policy is the average of the parameters of 10 learned policies. **(a)** FPOC based on human data from [9], which was used in this paper's previous experiments. **(b)** Exponential falloff of acuity. In this case, looking next to the target does not give reliable information about its presence, and so the learned policy prefers to look directly at the target. **(c)** A camera can locate objects reliably in its field of view, but not outside. The learned policy attempts to keep the object toward the edge of its field of view.

indeed people prefer to look near but not at visual targets.

C. Dependence on Visual System

So far, we have used the Foveal Peripheral Operating Curve (FPOC) shown in Figure 2c. This FPOC was first used by [9] and was based on careful psychophysical experiments. The I-POMDP framework allows us to investigate how an ideal oculomotor law may change if the FPOC of the sensory mechanism changes. This question is relevant to roboticists because robotic cameras do not typically have the same properties as a human eye. The question is also relevant to developmental scientists and clinicians that may study the development of visual search in infants and in adults with clinical eye conditions.

Here we considered two additional FPOCs. One is an exponential function that is sparser than the human FPOC: it has a sharp initial fall-off of acuity, but then has slightly higher acuity in the periphery (Figure 6b). The other is modeled after a standard camera with uniform acuity throughout its entire visual sensor and none elsewhere, resulting in a step-function FPOC (Figure 6c).

The resulting control laws are strikingly different from the original (Figure 6a), suggesting that the ideal VS strategy depends heavily on the specific FPOC of the visual system. This provides a warning against the usefulness of models of visual search derived from typical adults when these models are applied to guide robot cameras or to understand non-adult systems [2] for robots.

VI. CONCLUSION

We presented a model that addresses four critical limitation of current models of visual search: 1) Lack of formalization of how saccades are organized in time. 2) No mechanism for integrating information across fixations. 3) No account for how ideal behavior may change depending on the operating characteristics of the visual sensory system. 4) Lack of guarantees about optimality of the visual search laws.

The VS model of Geisler served as the starting point for a stochastic optimal control model of eye movement that integrates information across fixations. We showed that under this model the ideal visual search strategy depends critically on the operating characteristics of the visual sensory system. Even though a camera is not like an eye (it has good discrimination for a wider angle than the human fovea, but no peripheral discrimination) optimal scan paths can still be computed without peripheral information. This suggests promising extensions to the saliency approach, which currently considers all relevant information for planning eye-movements as contained in the current visual field.

REFERENCES

[1] N. J. Butko, L. Zhang, G. W. Cottrell, and J. R. Movellan, "Visual saliency model for robot cameras," in *International Conference on Robotics and Automation (ICRA)*, 2008.

[2] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 11, pp. 1254–1259, Nov 1998.

[3] A. M. Treisman and G. Gelade, "A feature-integration theory of attention," *Cognitive Psychology*, vol. 12, no. 1, pp. 97–136, Jan 1980.

[4] W. Kienzle, F. A. Wichmann, B. Schölkopf, and M. Franz, "A non-parametric approach to bottom-up visual saliency," in *Advances in Neural Information Processing Systems 19*, B. Schölkopf, J. Platt, and T. Hoffman, Eds. Cambridge, MA: MIT Press, 2007.

[5] A. Torralba, A. Oliva, M. S. Castelhano, and J. M. Henderson, "Contextual guidance of eye movements and attention in real-world scenes: The role of global features in object search," *Psychological Review*, vol. 113, no. 4, pp. 766–786, 2006.

[6] N. D. B. Bruce and J. K. Tsotsos, "Saliency based on information maximization," in *Advances in Neural Information Processing Systems 18*, Y. Weiss, B. Schölkopf, and J. Platt, Eds. Cambridge, MA: MIT Press, 2006, pp. 155–162.

[7] J. Harel, C. Koch, and P. Perona, "Graph-based visual saliency," in *Advances in Neural Information Processing Systems 19*, B. Schölkopf, J. Platt, and T. Hoffman, Eds. Cambridge, MA: MIT Press, 2007.

[8] L. Zhang, M. H. Tong, N. J. Butko, J. R. Movellan, and G. W. Cottrell, "A bayesian framework for dynamic visual saliency," (In Preparation), 2007.

[9] J. Najemnik and W. S. Geisler, "Optimal eye movement strategies in visual search," *Nature*, vol. 434, pp. 387–391, March 2005.

[10] L. P. Kaelbling, M. L. Littman, and A. R. Cassandra, "Planning and acting in partially observable stochastic domains," *Artificial Intelligence*, vol. 101, pp. 99–134, 1998.

[11] J. R. Movellan, "An infomax controller for real time detection of contingency," in *Proceedings of the International Conference on Development and Learning (ICDL)*, Osaka, Japan, 2005.

[12] J. Pineau, G. Gordon, and S. Thrun, "Point-based value iteration: an anytime algorithm for pomdps," in *IJCAI*, Acapulco, Mexico, 2003, pp. 1025–1032.

[13] J. Baxter and P. L. Bartlett, "Infinite-horizon policy-gradient estimation," *Journal of Artificial Intelligence Research*, vol. 15, pp. 319–350, November 2001.