High-Level and Generic Models for Visual Search: When does high level knowledge help?

A.L. Yuille
Smith-Kettlewell Eye Research Institute
2318 Fillmore Street
San Francisco, CA 94115
yuille@attila.ski.org

James Coughlan
Smith-Kettlewell Eye Research Institute
2318 Fillmore Street
San Francisco, CA 94115
coughlan@ski.org

Abstract

We analyze the problem of detecting a road target in background clutter and investigate the amount of prior (i.e. target specific) knowledge needed to perform this search task. The problem is formulated in terms of Bayesian inference and we define a Bayesian ensemble of problem instances. This formulation implies that the performance measures of different models depend on order parameters which characterize the problem. This demonstrates that if there is little clutter then only weak knowledge about the target is required in order to detect the target. However, at a critical value of the order parameters there is a phase transition and it becomes effectively impossible to detect the target unless high-level target specific knowledge is used. These phase transitions determine different regimes within which different search strategies will be effective. These results have implications for bottom-up and top-down theories of vision.

To appear in Proceedings of Computer Vision and Pattern Recognition: CVPR'99.

1. Introduction

Suppose we want to detect a target in background clutter. How much knowledge about the target do we need to use?

In recent work [8] Yuille and Coughlan quantified the difficulty of detecting targets, such as roads, in images in terms of an *order parameter* K. This order parameter was defined in terms of the statistical properties of the images and the target and was used to characterize both the *expected error* in the solution and the *expected convergence rate*. It was shown that the detection task had a phase transition at $K = 0$. For $K < 0$ it was impossible, on average, to detect the target by any algorithm. For $K > 0$ detection is possible. More precisely, the detection task is formulated as MAP estimation and is equivalent to maximizing an appropriate *reward function*. We proved that the expected number of *ghosts* (false paths caused by random alignments of off-road pixels) with higher reward than the *true path reward* decreases as 2^{-KN} where N is the length of the road. Moreover, the expected error rate and the algorithmic convergence time decrease exponentially with K. The size of K increases with the effectiveness of the edge detector and with the greater the amount of prior knowledge available about the target. The results were obtained by mathematical proofs on the Bayesian ensemble of target detection problem instances using the formulation of road tracking by Geman and Jedynak [5].

In this paper, we use similar techniques to explore a related problem. How much harder do we make target detection by using a weaker model (i.e. a weaker prior probability distribution)? Can we quantify how much easier we make the task by using more information about the target?

2. Bottom-Up and Top-Down Vision

But why should we want to use a weaker prior? Firstly, there may not be enough information about the target to have an accurate prior for it (or it would cost too much to get this knowledge). Secondly, we may want to search for several different targets and it would seem more economical to use one prior model which would account for all of these targets (at the cost of modelling each of them relatively poorly) rather than having different models for each target. Thirdly, algorithmic considerations may favour using a weaker prior rather than a prior which is more accurate but harder to compute with.

We also observe that the standard bottom-up paradigm of computer vision can be thought of in these terms. The early processing of the images makes use of general purpose weak prior assumptions (which often seem so intuitive that they are unstated) such as edges being smooth.

For example, a recent system by Geiger and Liu [4] for detecting human figures uses weak prior knowledge of edge smoothness, as implemented by snakes [6], to detect boundary segments of objects which are then grouped, using object specific knowledge, to locate specific objects. This system is effective and computationally efficient but it will break down if the background clutter is sufficiently complicated that the snake algorithms are unable to find the object boundaries. In such situations, however, it *may still be possible to detect the object if a high-level prior model is used*. It is well known that humans can recognize certain objects, such as Mooney faces and dalmatian dogs (see, [7]), when it appears that low level assumptions about the domain will break down. In such cases, only a top-down strategy seems possible. But a purely top-down strategy is probably more computationally intensive than a bottom-up strategy. When can one get away with using an efficient bottom-up strategy?

The debate between bottom-up and top-down paradigms for vision is of long standing. See the chapter by Mumford in [7] for a recent discussion of these issues. But what has been lacking so far is a concrete quantitive analysis of the relative effectiveness of these strategies.

This paper develops a theory to contrast bottom-up and top-down approaches for the specific problem of road tracking [5]. This is a domain which is simple enough to allow rigorous mathematical analysis but is realistic enough for the results to be directly generalizable to the situations studied by [4],[1], and hopefully to other domains. It helps that the Geman and Jedynak theory for road tracking [5] is both theoretically elegant and also highly effective in practice.

More precisely, our approach characterizes the road detection task in terms of order-parameters for both the bottom-up and top-down strategies. It will be shown that in certain *regimes* of the order parameters both strategies will be effective (in the sense of finding a close approximation to the true solution). In other regimes the top-down strategy will work but the bottom-up strategy will not. In yet other regimes the target detection problem becomes impossible to solve by any approach. We will also analyze the speed of algorithms for solving the problems. Our results suggest that the bottom-up strategy can often be significantly quicker than the top-down strategy and it makes sense to use it, particularly if we are searching for one of several different types of road.

2.1. Background and Previous Work

Tracking curved objects in real images is an important practical problem in computer vision. We consider a specific formulation of the problem of road tracking from aerial images by Geman (D.) and Jedynak [5]. This approach assumes that both the intensity properties and the geometrical shapes of the target path (i.e. the edge contour) can be modelled statistically. This path can be considered to be a set of elementary path segments joined together. We first consider the intensity properties along the edge and then the geometric properties.

The image properties of segments lying on the path are assumed to differ, in a statistical sense, from those off the path. More precisely, we can design a filter $\phi(.)$ with output $\{y_x = \phi(I(x))\}$ for a segment at point x so that:

$$P(y_x) = P_{on}(y_x), \;\; if \text{ ``}x'' \, lies \, on \, the \, true \, path$$
$$P(y_x) = P_{off}(y_x), \;\; if \text{ ``}x'' \, lies \, off \, the \, true \, path. \quad (1)$$

For example, we can think of the $\{y_x\}$ as being values of the edge strength at point x and P_{on}, P_{off} being the probability distributions of the response of $\phi(.)$ on and off an edge. The set of possible values of the random variable y_x is the *alphabet* with *alphabet size* M.

We now consider the geometry of the target contour. We require the path to be made up of connected segments $x_1, x_2, \ldots, x_N$. There will be a Markov probability distribution $P_g(x_{i+1}|x_i)$ which specifies prior probabilistic knowledge of the target. Each point x has a set of Q neighbours. Following terminology from graph theory, we refer to Q as the *branching factor*. We will assume that the distribution P_g depends only on the relative positions of x_{i+1} and x_i. In other words, $P_g(x_{i+1}|x_i) = P_{\Delta g}(x_{i+1} - x_i)$. An important special case is when the probability distribution is uniform for all branches (i.e. $P_{\Delta g}(\Delta x) = U(\Delta x) = 1/Q$ $\forall \Delta x$).

By standard Bayesian analysis, the optimal (MAP estimate) path $X^* = \{x_1^*, \ldots, x_N^*\}$ maximizes the sum of the log posterior ratios:

$$R(X) = \sum_i \log \frac{P_{on}(y_{(x_i)})}{P_{off}(y_{(x_i)})} + \sum_i \log \frac{P_{\Delta g}(x_{i+1} - x_i)}{U(x_{i+1} - x_i)}, \quad (2)$$

where the sum i is taken over all points on the target. $U(x_{i+1} - x_i)$ is the uniform distribution and its presence merely changes the log posterior $R(X)$ by a constant value. It is included to make the form of the intensity and geometric terms similar, which simplifies our later analysis.

We will refer to $R(X)$ as the *reward* of the path X which is the sum of the *intensity rewards* $\log \frac{P_{on}(y_{(x_i)})}{P_{off}(y_{(x_i)})}$ and the *geometric rewards* $\log \frac{P_{\Delta g}(x_{i+1} - x_i)}{U(x_{i+1} - x_i)}$.

As we will show, this formulation can be extended to deal with second order and higher level priors (provided they are shift-invariant). (See figures (2,3,4, 5) for examples of first order models and figure (6) for examples of second order Markov models.) This allows our theory to apply to models such as snakes [6]. (It is straightforward to transform the standard energy function formulation of snakes into a Markov chain by discretizing and replacing the derivatives by differences. The smoothness constraints, such as membranes and thin plate terms, will transform into first and second order Markov chain connections respectively). Recent work by Zhu [10] shows that Markov chain models of this type can be learnt using Minimax Entropy Learning theory from a representative set of examples.

3. Fundamental Limits

As shown in [8], the Bayesian formulation of our problems *naturally gives rise to a probability distribution on the ensemble of problem instances*, which is called the *Bayesian Ensemble*. Using the Bayesian Ensemble we can compute *order parameters* which determine the behaviour of typical problem instances (i.e. those which occur with high probability). Technically, the proofs involve adapting techniques from information theory, such as Sanov's theorem, which were developed to bound the probability of rare events occurring [3]. For the road tracking problem, a rare event would be when a subpath in the background noise/clutter has greater reward than a subpath of the true road – i.e. looks more like a road.

For the road tracking task, the *order parameter* K which is the sum of (twice) the Bhattacharyya distances $2B(P_{on}, P_{off}) = -2\log\{\sum_y P_{on}^{1/2}(y)P_{off}^{1/2}(y)\}$ and $2B(P_{\Delta g}, U)$ minus the entropy $H(U)$ of the uniform distribution. The smaller these two distances the harder the problem becomes. Intuitively, the better the edge detector – as evaluated by $2B(P_{on}, P_{off})$ – and the more specific the signal – as measured by $2B(P_{\Delta g}, U) - H(U)$, the easier it is to detect. Yuille and Coughlan [8] show that there is a phase transition at $K = 0$ so that the problem becomes effectively impossible to solve *by any algorithm* for $K < 0$. Other properties of interest to the problem, such as the expected convergence rates of algorithms and the expected errors in the estimated road position can be expressed in terms of K, see section (7).

All these results, however, assumed that there was a fixed geometric prior model. What happens if we use the wrong prior for the reasons specified in the introduction? How badly do our results degrade if we use the wrong prior? Suppose we have several types of roads and we do not know which one we should be looking for – can we get away with using a simple generic prior to detect these different types of road simultaneously? We now address these issues.

4. High-Level and Generic Models

Suppose we have a single high-level model for a road with a *high level* geometric prior $P_H(\Delta x)$. Let us assume a weaker *generic* prior $P_G(\Delta x)$. We can define two different rewards R_G and R_H:

$$R_G(\{x_i\}) = \sum_i \log \frac{P_{on}(y_i)}{P_{off}(y_i)} + \sum_i \log \frac{P_G(\Delta x_i)}{U(\Delta x_i)},$$
$$R_H(\{x_i\}) = \sum_i \log \frac{P_{on}(y_i)}{P_{off}(y_i)} + \sum_i \log \frac{P_H(\Delta x_i)}{U(\Delta x_i)}. \quad (3)$$

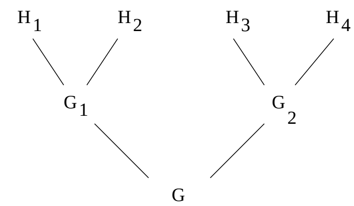

Figure 1. The Hierarchy. Two high-level models H_1, H_2 "project" onto a low-level generic model P_{G_1}. In situations with limited clutter it will be possible to detect either H_1 or H_2 using the single generic model P_{G_1}. This idea can be extended to have hierarchies of projections. This is analogous to the superordinate, basic level, and subordinate levels of classification used in cognitive psychology.

The optimal Bayesian strategy to search for the road would be to use the high level model and evaluate paths based on their rewards R_H. But this strategy ignores the computation time involved in using the prior P_H. For example, P_H might be a second or higher order model while P_G might be a first order Markov model (which would be easier to search over). Also, we might not know the exact form of P_H. Perhaps the most important situation, to be considered in a later section, is when we can use a single generic model to search for a target which may be one of several different models. Using a single generic model (provided it is powerful enough) to detect the road can be significantly faster than testing each possible road model in turn.

But how "weak" should the generic prior P_G be? One possibility is that P_G is an approximation to P_H. Such a situation will often arise when we do not know the true prior distribution of a target. If P_G is just a minor perturbation of P_H then standard analysis shows that the concavity of the Bayes risk means the system will be stable to such perturbations. A more important case arises when P_G *is a poor approximation to* P_H. In what regimes can we get away with using a poor approximation? We will give results for this case.

A more interesting form of "weakness" is when the generic prior P_G is a projection of the high-level prior P_H onto a simpler class of probability distributions. This allows us to formulate the idea of a hierarchy in which the priors for several high-level objects would all project onto the identical low-level prior, see figure (1). For example, we might have a set of priors $\{P_{H_i} : i = 1, ..., M\}$ for different members of the cat family. There might then be a generic prior P_G onto which all the $\{P_{H_i}\}$ project and which is considered the embodiment of "cattiness". See [2] for details on this projection approach and how it relates to the Zhu, Wu, Mumford theory of learning [9].

In this paper we will be concerned with the *Amari* (Am) and *Bhatta* (Bh) projections. These projections imply that the distributions to be projected, $P_H(x)$, are related to the generic distributions $P_G(x)$ by:

$$\sum_x P_H(x) \log P_G(x) = \sum_x P_G(x) \log P_G(x), \ Am$$

$$\frac{\sum_x P_H(x) P_G^{-1/2}(x) \log P_G(x)}{\sum_x P_G^{1/2}(x) \log P_G(x)} = \frac{\sum_x P_H(x) P_G^{-1/2}(x)}{\sum_x P_G^{1/2}(x)} \ Bh.$$

These two projections are special in that they allow us to obtain analytic expressions for the order parameters. It should be emphasized that *order parameters can be derived for other projections or, as will show, for other forms of approximations*, but these parameters are expressed in terms of the minimization of a function of one variable. Such order parameters are easy to calculate by computer but do not have the simple intuitive forms which arise from the Amari and Bhatta projections.

The Amari projection is important because it corresponds to the minimax entropy criterion used in the learning theory developed by Zhu, Wu, Mumford [9]. It allows us to obtain analytic expressions for the order parameters for a specific road search task. The Bhatta projection is slightly more complicated but it gives us analytic expressions for the order parameters for a slightly more realistic road search task.

5. The Order Parameters

We use the techniques outlined in section (3) to calculate the order parameters. These calculations are variants of those used in [8] and are too lengthy to include here. (The details are written up in a technical report which can be obtained upon request).

Our first case is when the generic prior P_G is obtained from a high-level prior P_H by Amari projection. For reasons of space, we also use this example to illustrate what happens if we use a "poor" approximation to the true high-level model. We consider using the generic reward R_G and the high-level reward R_H. The criterion is to determine the probability that *a ghost* (a fluctuation of off-road pixels) will have higher reward than the *expected true path reward* under either of the two rewards. We obtain order parameters

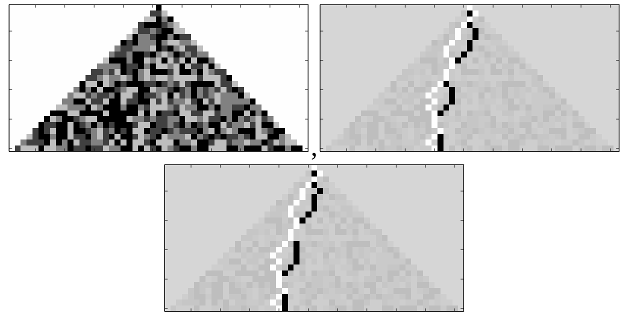

Figure 2. The Ultra Regime $K_H^G < K_G^A < 0$. Top left, the input image. Top right, the true path is shown in white and the *errors* of the best path found using the Generic model are shown in black. Bottom left, similar, for the High-Level model. Observe that although the best paths found are close to the true path there is comparatively little overlap. A dynamic programming algorithm was used to determine the best solution for either choice of reward.

K_H^A, K^A, G for the high-level and generic rewards respectively (the superscript A refers to Amari):

$$K_H^A = D(P_{on}||P_{off}) + D(P_H||U) - \log Q,$$
$$K_G^A = D(P_{on}||P_{off}) + D(P_G||U) - \log Q. \quad (4)$$

It follows from the definition of Amari projection that $K_H^A - K_G^A = D(P_H||U) - D(P_G||U) = D(P_H||P_G)$ (where $D(P||Q) = \sum_y P(y) \log P(y)/Q(y)$ is the *Kullback-Leibler* divergence between distributions $P(y)$ and $Q(y)$). Therefore the high-level prior P_H has a bigger order parameter by an amount which depends on the distance between it and P_G as measured by the Kullback-Leibler divergence $D(P_H||P_G)$. Recall [8] that the target detection problem becomes insolvable (by any algorithm) when the order parameter is less than zero. Hence there are three regimes: (I) The *Ultra Regime*, see figure (2), is when $K_G^A < K_H^A < 0$ (i.e. $D(P_H||U) + D(P_{on}||P_{off}) < \log Q$) and the problem cannot be solved (on average) by any model (or algorithm). (II) The *Challenging Regime*, see figure (3), where $K_G^A < 0 < K_H^A$ (i.e. $\log Q < D(P_H||U) + D(P_{on}||P_{off}) < \log Q + D(P_H||P_G)$) within which the problem can be solved by the high-level model but not by the generic model. (III) The *Easy Regime*, see figure (4), where $K_H^A > K_G^A > 0$ and the problem can be solved by either the generic or the high-level model.

In our simulations, see figures (2,3,4), we *generate* the target true paths by *stochastic sampling from the high level model*. To *detect* the best path we apply a dynamic programming algorithm to optimize the high-level or generic

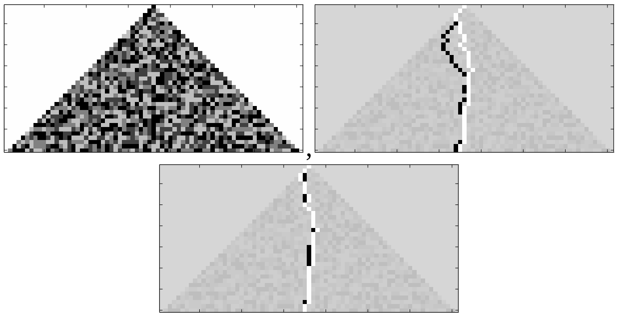

Figure 3. The Challenging Regime $K_G^A < 0 < K_H^A$. Same conventions as previous figure. Observe that the Generic models fails (top right) but the High-Level model succeeds (bottom).

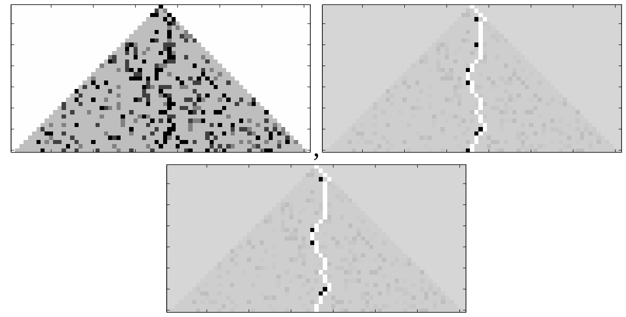

Figure 4. The Easy Regime $0 < K_G^A < K_H^A$. Same conventions as previous figure. In this regime both the Generic and High-Level models succeed.

reward functions applied to the generated data. Dynamic programming is guaranteed to find the solution with highest reward.

The second case is when we use the Bhatta projection with rewards either R_G or R_H. We now use the criterion that a ghost will have higher reward than the true path (i.e. we now take the fluctuations of the true path rewards into account which is more realistic). This yields order parameters:

$$K_H^B = 2B(P_{on}||P_{off}) + 2B(P_H, U) - \log Q$$
$$K_G^B = 2B(P_{on}||P_{off}) + \hat{B}(P_H, P_G) - \log Q, \quad (5)$$

where $B(P_{on}||P_{off}) = -\log \sum_y \{P_{on}^{1/2}(y) P_{off}^{1/2}(y)\}$ and $\hat{B}(P_G, P_H) = -\log(\sum_x P_G^{1/2}(x))(\sum_x P_H(x) P_G^{-1/2}(x))/Q$. It can be verified that $2B(P_H, U) > \hat{B}(P_G, P_H)$ and so, once again, we will have three similar regimes – Ultra, Challenging, and Easy – corresponding to $K_G^B < K_H^B < 0$, $K_G^B < 0 < K_H^B$, and $K_H^B > K_G^B > 0$. These regimes have the same interpretations as above.

The third case is when we do not use the Amari or Bhatta projections. This includes: (a) projections differing from Amari and Bhatta, and (b) situations where P_G is a (poor) approximation to P_H. This gives order parameters:

$$K_H = 2B(P_{on}||P_{off}) + 2B(P_H, U) - \log Q$$
$$K_G = \sum_{i=1}^{4} \log Z_i[\gamma^*] - \log Q, \quad (6)$$

where $Z_1[\gamma] = \sum_y P_{on}^{1-\gamma}(y) P_{off}^{\gamma}(y)$, $Z_2[\gamma] = \sum_x P_H(x) P_G^{-\gamma}(x)$, $Z_3[\gamma] = \sum_y P_{on}^{\gamma}(y) P_{off}^{1-\gamma}(y)$, and $Z_4[\gamma] = \sum_x P_G^{\gamma}(x)$. The value $\gamma^* = \arg\min_{0 \leq \gamma \leq 1} \sum_{i=1}^{4} \log Z_i[\gamma]$. ($\log Z[\gamma]$ is convex so minimizing it is easy).

Once again, we get the same three regimes with the same interpretations.

In our experience, changing the projections (approximations) only causes the boundaries of the regimes to shift and does not alter their basic properties. Moreover, so far, we have observed little change in the regime boundaries if we switch from the Amari to the Bhatta projection (see technical report).

6. Multiple Hypotheses and Higher-Order Markov Models

We now apply our theory to deal with multiple (two or more) high-level models and with high-level models defined by second-order Markov chains.

The prototypical case for two, or more, high-level models is illustrated in figure (5). High-level model H_1 prefers roads which move to the right (see the white paths in the left hand panels of figure (5)) while high-level model H_2 likes roads moving to the left (see white paths in the right panels). Both models H_1 and H_2 *project to the same generic model* G, by Amari projection, and thus form part of a hierarchy, see figure (1). Our theory again enables us to calculate order parameters and identify three regimes: (I) The Ultra Regime where none of the models (H_1, H_2 or G) can find the target. (II) The Challenging Regime where the high-level models H_1, H_2 can find targets generated by H_1 and H_2 *respectively* but the generic model G cannot find either. (III) The Easy Regime where all the models can locate the targets effectively. Once again, the best paths for the different rewards was found using dynamic programming (which is guaranteed to find the global solution).

In the Easy Regime, little is gained by using the two

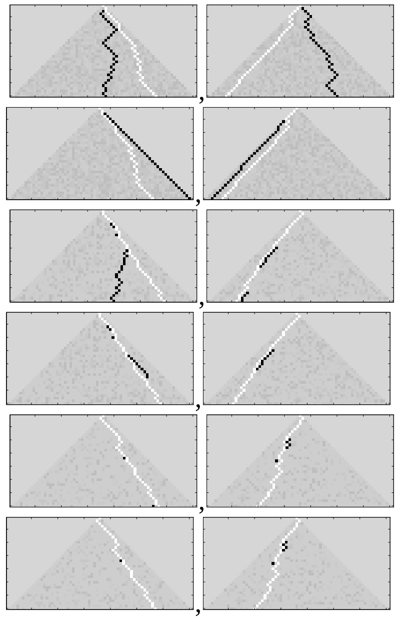

Figure 5. Two High-Level models H_1, H_2. Three sets of four panels for Ultra, Challenging, and Easy regimes (top to bottom). The data in the left and right columns is generated by H_1 and H_2 respectively. The top two rows are the Ultra regime with the true paths (white) and the errors of the best paths (black) for Generic model (first row), H_1 model (second row, left) and H_2 model (second row, right). All models give poor results in this regime. The middle two rows are similarly organized and show good results for the High-Level models and significantly poorer results for the Generic. The bottom two rows (same conventions) demonstrate the effectiveness of all models in the Easy regime.

high-level models. It may indeed be more computationally efficient to use the generic model to detect the target. The target could then be classified as being H_1 or H_2 in a subsequent classification stage. We will discuss computational tradeoffs of these two approaches in the next section.

We now repeat this example using high-level models H_3, H_4 defined by second order Markov chains, see figure (6). This second order property allows us to obtain more interesting models. For example, model H_3 generates very wiggly roads ("English" roads) (see left panel of figure (6)) while model H_4 generates roads that have long straight sections with occasional sharp changes in direction ("Roman" roads, see right hand panels). It is straightforward to compute order parameters for these models (the second-order Markov property requires slight modifications to the earlier calculations) and, as before, we get order parameters which specify the three standard Ultra, Challenging, and Easy regimes – see figure (6). In this figure, we point out a fluke where the high-level model H_4 correctly found the target even in the Ultra Regime. By our theory, this is possible though highly unlikely. Another unlikely outcome is shown in the bottom right panel where the H_4 model has detected the target to *one hundred percent accuracy*. This is reflected in the overall darkness of the panel because, with no black pixels to indicate errors, our graphics package has altered the brightness of the panel (compared to the other panels which do contain black errors). Dynamic programming is used to find the best solutions by global optimization.

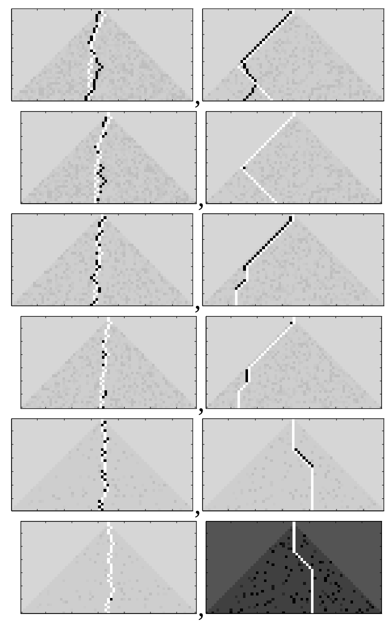

Figure 6. Two High-Level models second-order Markov models H_3, H_4. Three sets of four panels for Ultra, Challenging, and Easy regimes (top to bottom). The data in the left and right columns is generated by H_3 and H_4 respectively. The top two rows are the Ultra regime with the true paths (white) and the errors of the best paths (black) for Generic model (first row), H_3 model (second row, left) and H_4 model (second row, right). All models give poor results in this regime. The middle two rows are similarly organized and show good results for the High-Level models and significantly poorer results for the Generic. The bottom two rows (same conventions) demonstrate the effectiveness of all models in the Easy regime.

7. More Precise Analysis of Performance Measures

So far, we have only discussed whether the target can be detected or not in terms of the order parameters of the models. But our analysis also yields more precise, performance measures which determine the accuracy of the solution and the complexity of A* search algorithms for finding the solution. These results (see technical report for more details) suggest that these performance measures are very sensitive to the precise values of the order parameters K close to the phase transitions at $K = 0$. But they are relatively insensitive to the precise values of K elsewhere. (This is consistent with the typical behaviour of physical systems which have phase transitions).

8. Summary and Conclusions

This paper investigated how much prior knowledge is needed to detect a target road in the presence of clutter. We used the concept of order parameters to determine whether a target could be detected using a general purpose "generic" model or whether a more specific high level model was needed. At critical values of the order parameters the problem becomes unsolvable without the addition of extra prior knowledge. We discussed the implication of these results for bottom-up and top-down theories of vision.

The results of this paper were obtained by analysis of the Bayesian ensemble of problem instances. We anticipate that our approach will generalize to other vision problems and can be used to determine performance measures for models in terms of order parameters. Our results should be of particular importance to problems for which one can use learning techniques [9] to determine suitable probability models as input to our analysis.

Acknowledgments

We want to acknowledge funding from NSF with award number IRI-9700446, from the Center for Imaging Sciences funded by ARO DAAH049510494,from the Smith-Kettlewell core grant, and the AFOSR grant F49620-98-1-0197 to ALY. We thank Mario Ferraro for asking interesting questions which inspired some of this work. Scott Konishi and Song Chun Zhu gave useful feedback.

References

[1] James M. Coughlan, A.L. Yuille, D. Snow and C. English. "Efficient Optimization of a Deformable Template Using Dynamic Programming". In *Proceedings Computer Vision and Pattern Recognition. CVPR'98*. Santa Barbara. California. 1998.

[2] James M. Coughlan and A.L. Yuille. "A Phase Space Approach to Minimax Entropy Learning". In *proceedings NIP98*. 1998.

[3] T.M. Cover and J.A. Thomas. **Elements of Information Theory**. Wiley Interscience Press. New York. 1991.

[4] D. Geiger and T-L Liu. "Top-Down Recognition and Bottom-Up Integration for Recognizing Articulated Objects". In **EMMCVPR'97**. Ed. M. Pellilo and E. Hancock. Springer-Verlag. CS 1223. 1997.

[5] D. Geman. and B. Jedynak. "An active testing model for tracking roads in satellite images". *IEEE Trans. Patt. Anal. and Machine Intel.* Vol. 18. No. 1, pp 1-14. January. 1996.

[6] M. Kass, A. Witkin, and D. Terzopoulos. "Snakes: Active Contour models". In *Proc. 1st Int. Conf. on Computer Vision*. 259-268. 1987.

[7] D.C. Knill and W. Richards. (Eds). **Perception as Bayesian Inference**. Cambridge University Press. 1996.

[8] A.L. Yuille and J.M. Coughlan. "Convergence Rates of Algorithms for Visual Search: Detecting Visual Contours". In *Proceedings NIPS'98*. 1998.'

[9] S-C Zhu, Y-N Wu and D. Mumford. FRAME: Filters, Random field And Maximum Entropy. Int'l Journal of Computer Vision 27(2) 1-20, March/April. 1998.

[10] S.C. Zhu. "Embedding Gestalt Laws in Markov Random Fields". Submitted to *IEEE Computer Society Workshop on Perceptual Organization in Computer Vision*.