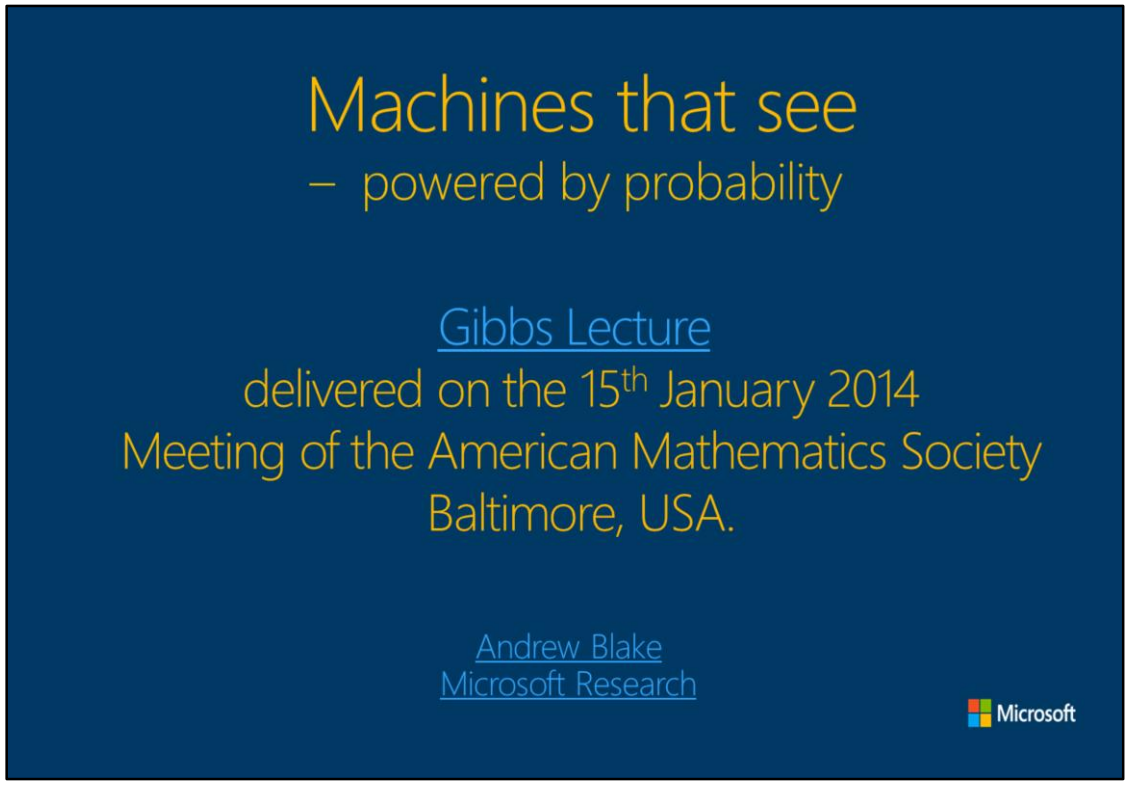
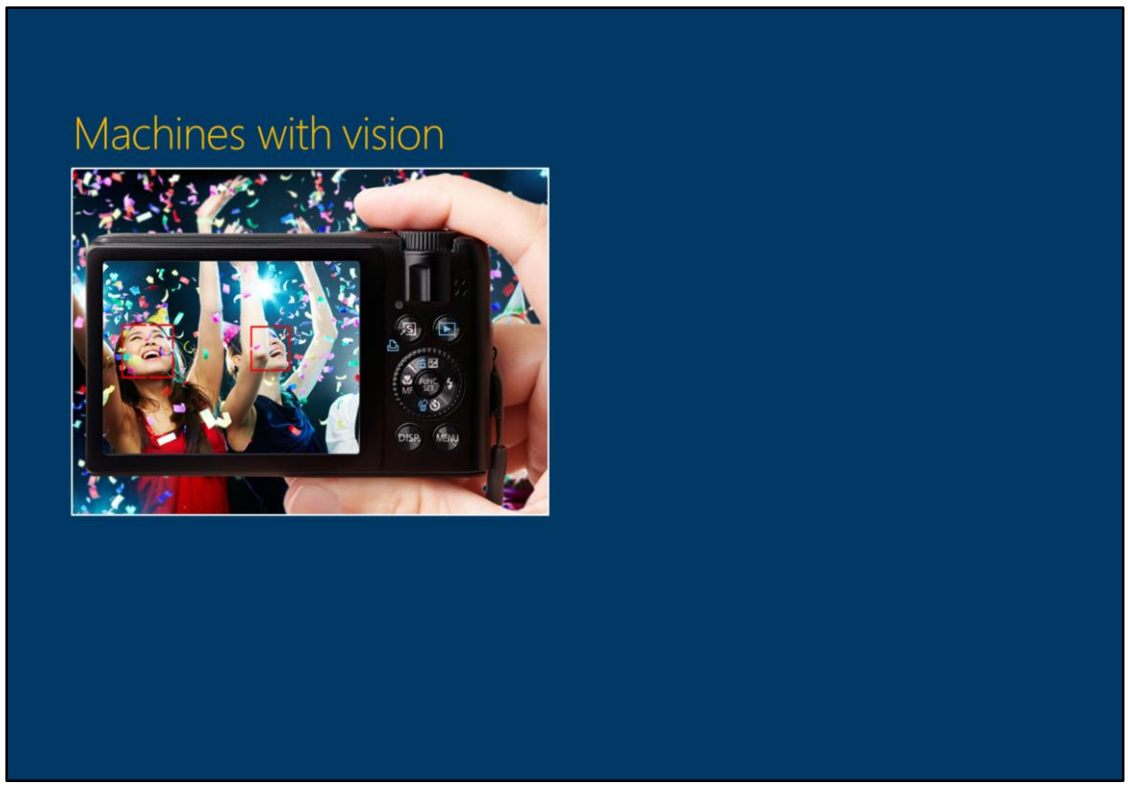


Gibbs Lecture

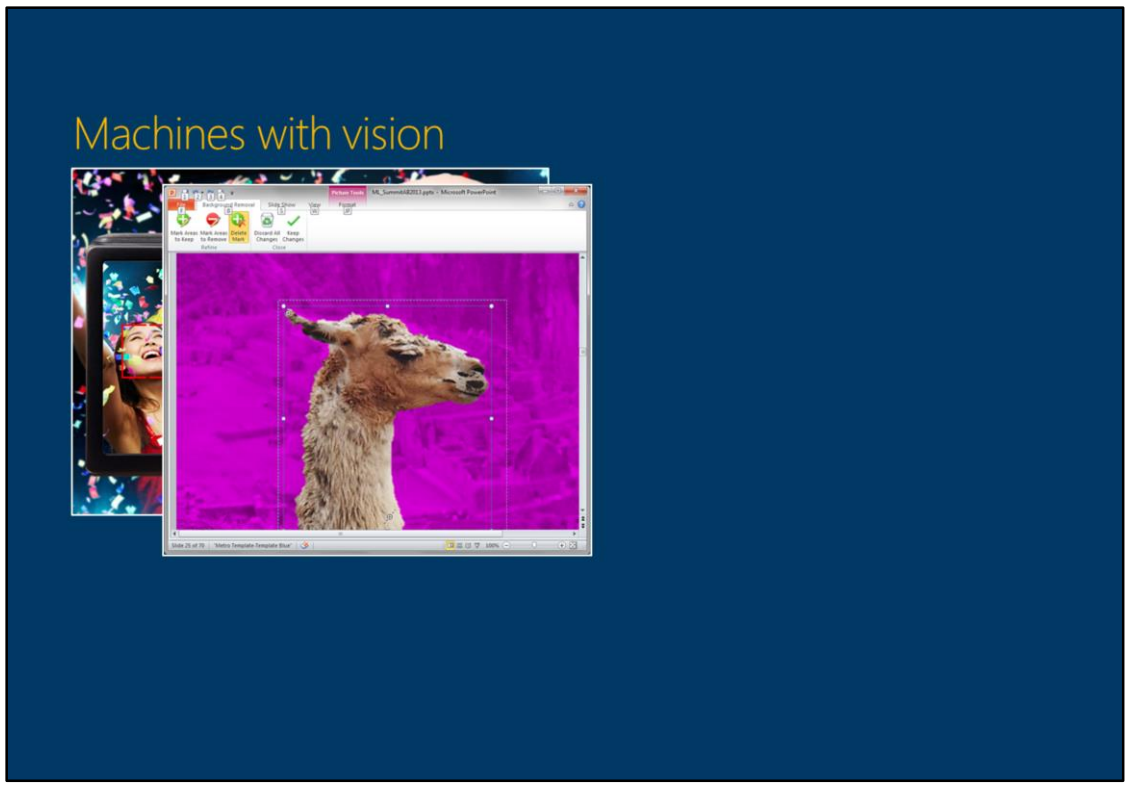
ABSTRACT. Machines with some kind of ability to see have become a reality in the last decade, and we see vision capabilities in cameras and photography, cars, graphics software and in the user interfaces to appliances. Such machines bring benefits to safety, consumer experiences, and healthcare, and their operation is based on mathematical ideas.

The visible world is inherently ambiguous and uncertain so estimation of physical properties by machine vision often relies on probabilistic methods. Prior distributions over shape can help significantly to make estimators for finding and tracking objects more robust. Learned distributions for colour and texture are used to make the estimators more selective. These ideas fit into a "generative" philosophy of vision as inference: exploring hypotheses about the contents of a scene that explain an image as fully as possible. More recently this explanatory approach has partly given way to powerful, direct, "discriminative" estimation methods, whose operating parameters are learned from large data sets. It seems likely that the most capable vision systems will come ultimately from some kind of fusion of the generative and discriminative approaches.

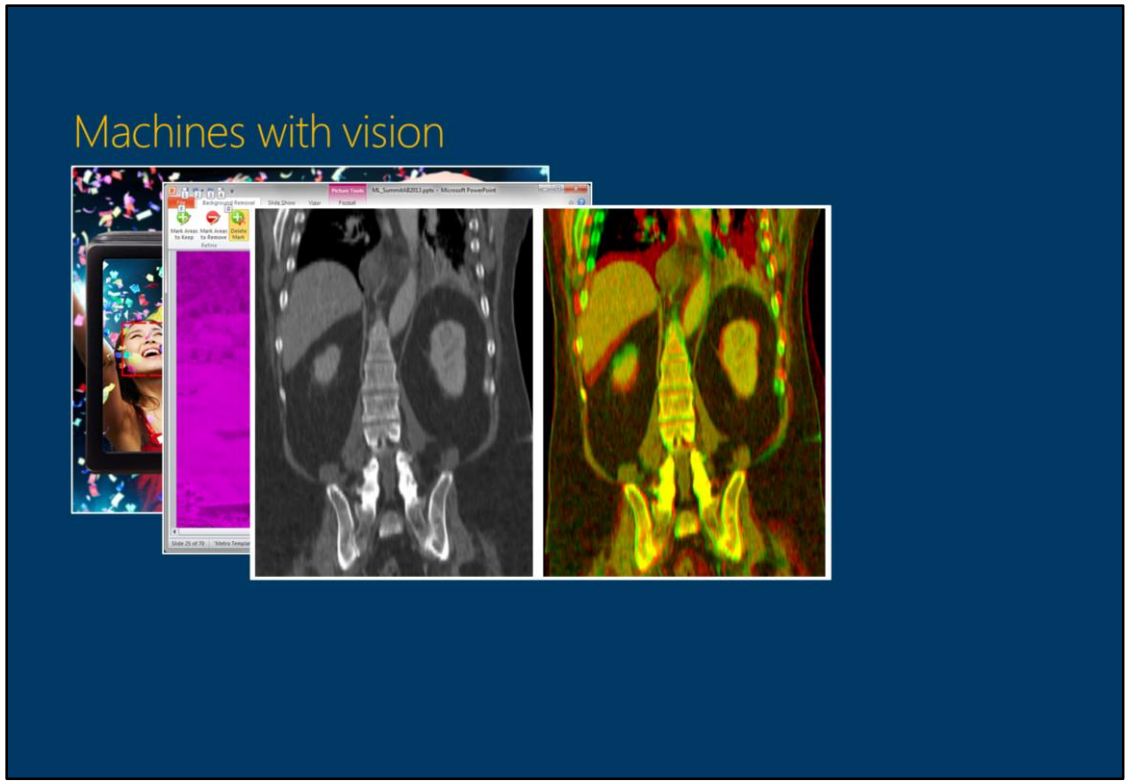
Andrew Blake, Microsoft Research.



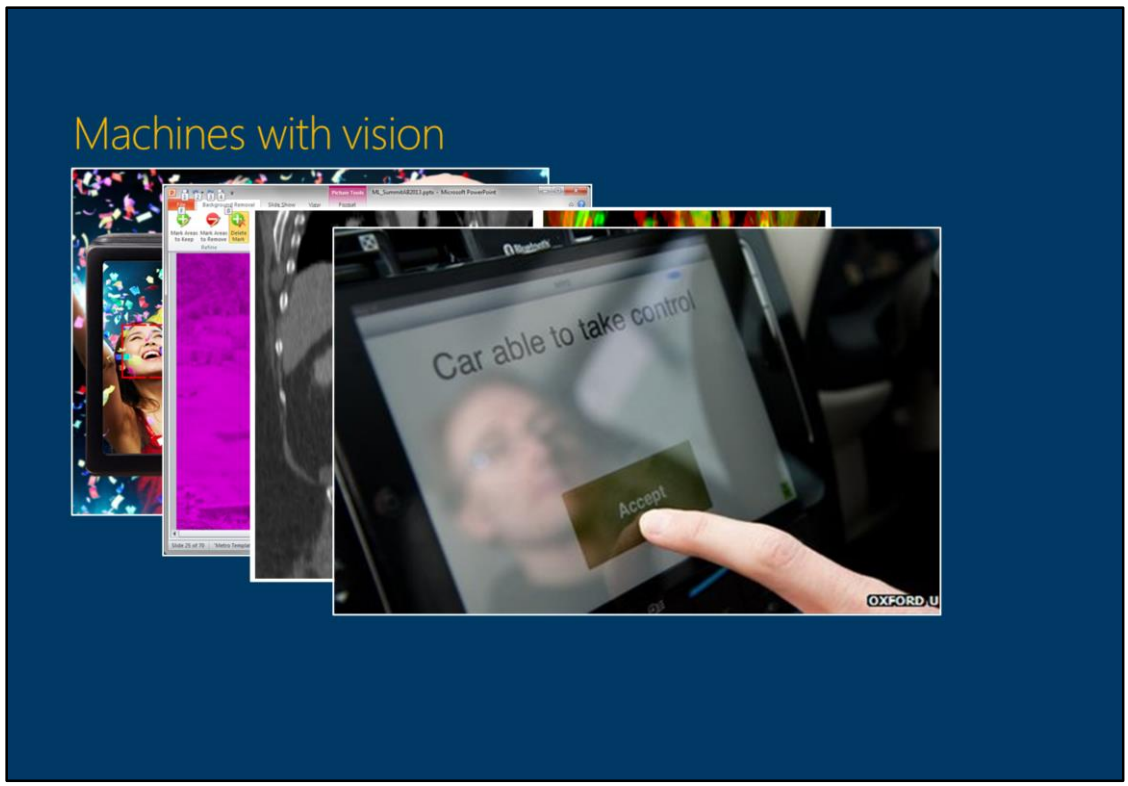
First, we can celebrate the fact that there are a number of commercial products that have reached a significant scale which embody machine vision. One widespread success has been the incorporation of face recognition in consumer devices – cameras and phones. Here a mass market compact camera is shown that uses face recognition to help compose a good picture, and this facility became widespread in the early 2000s.



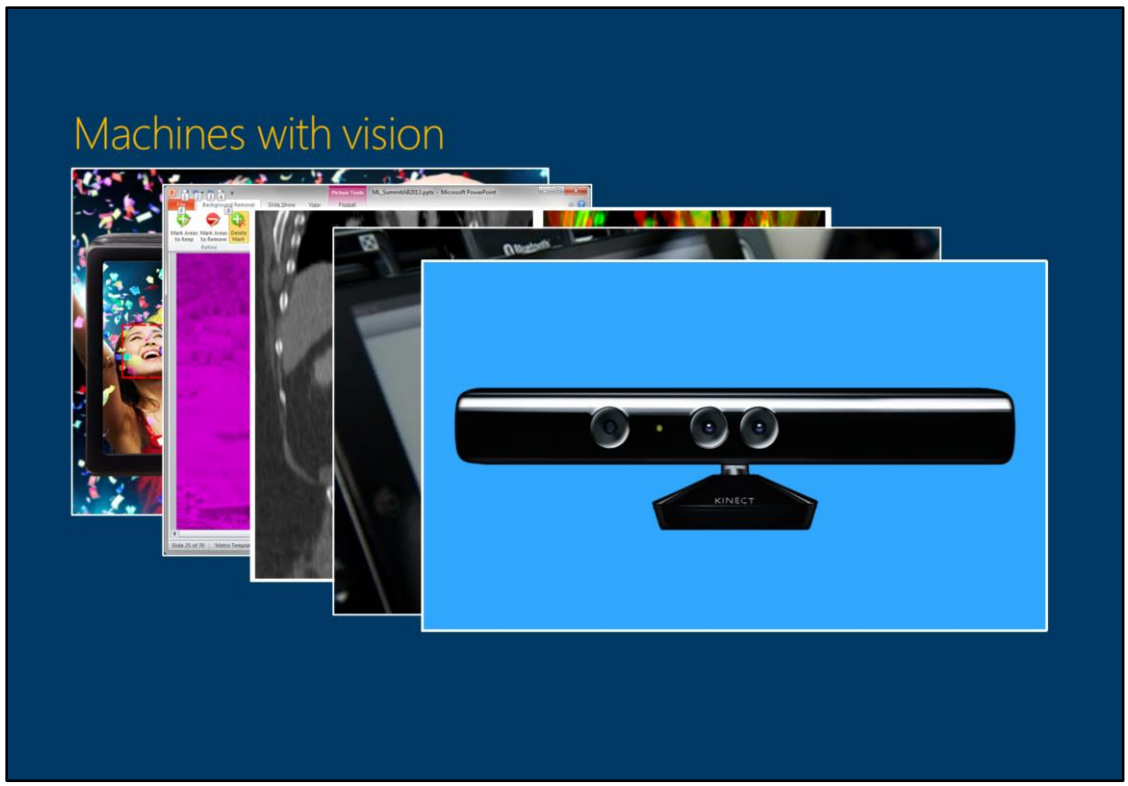
Another application of machine vision that has hundreds of millions of customers is automatic video editing. Here the background removal facility in Microsoft Office is illustrated, that allows a non-expert user to peel a subject away from its background, so that it can be placed in a new setting. Techniques derived from machine vision allow this to happen with a high degree of automation and minimal input from the user, who needs simply to make one or two mouse strokes to achieve the result.



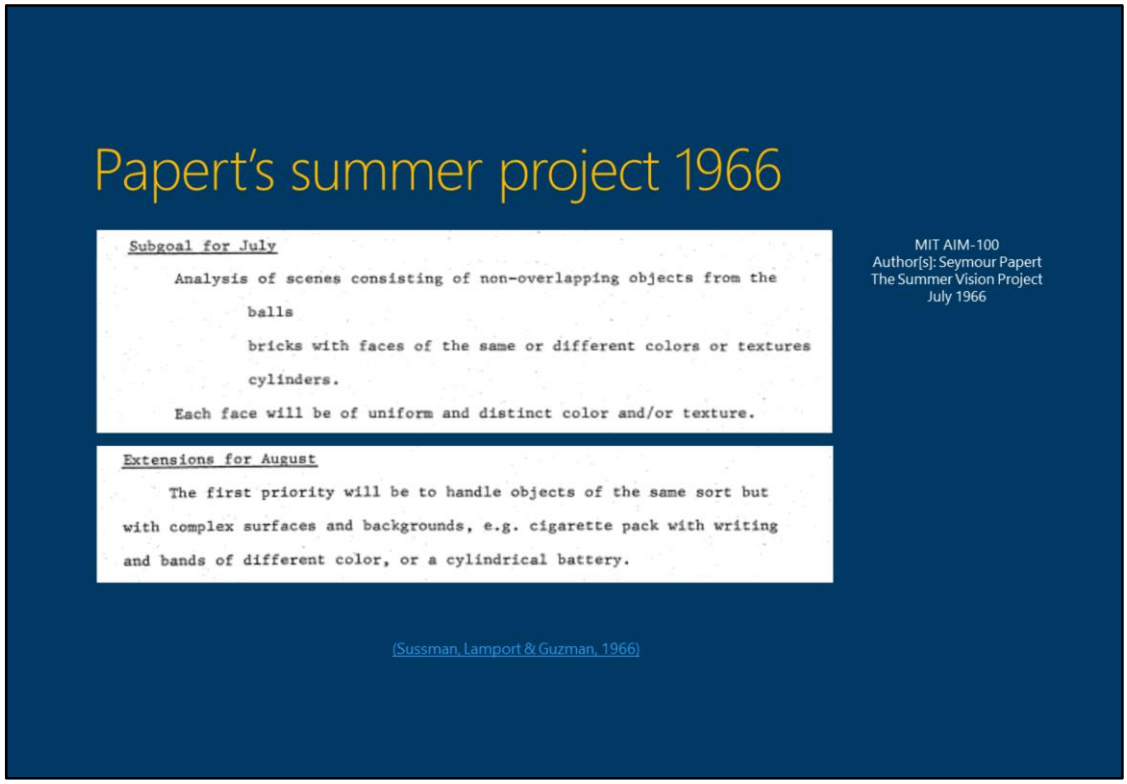
Machine vision techniques have been very influential in medical diagnosis. The figure illustrates a system for registration of two diagnostic images – here of the lungs – taken on different occasions. The human body being flexible, there is significant non-rigid deformation of the images between sessions. Registering the images in this way is important for making clear comparisons between images. An algorithm capable of achieving good registration must be able to estimate the inter-image deformation.



There has been a good deal of publicity in the last decade about breakthroughs with autonomous vehicles. They use a variety of sensors and GPS mapping to navigate automatically and avoid collisions. Some use active laser sensors to detect obstacles and landmarks, but the system illustrated from the University of Oxford uses cameras with machine vision.

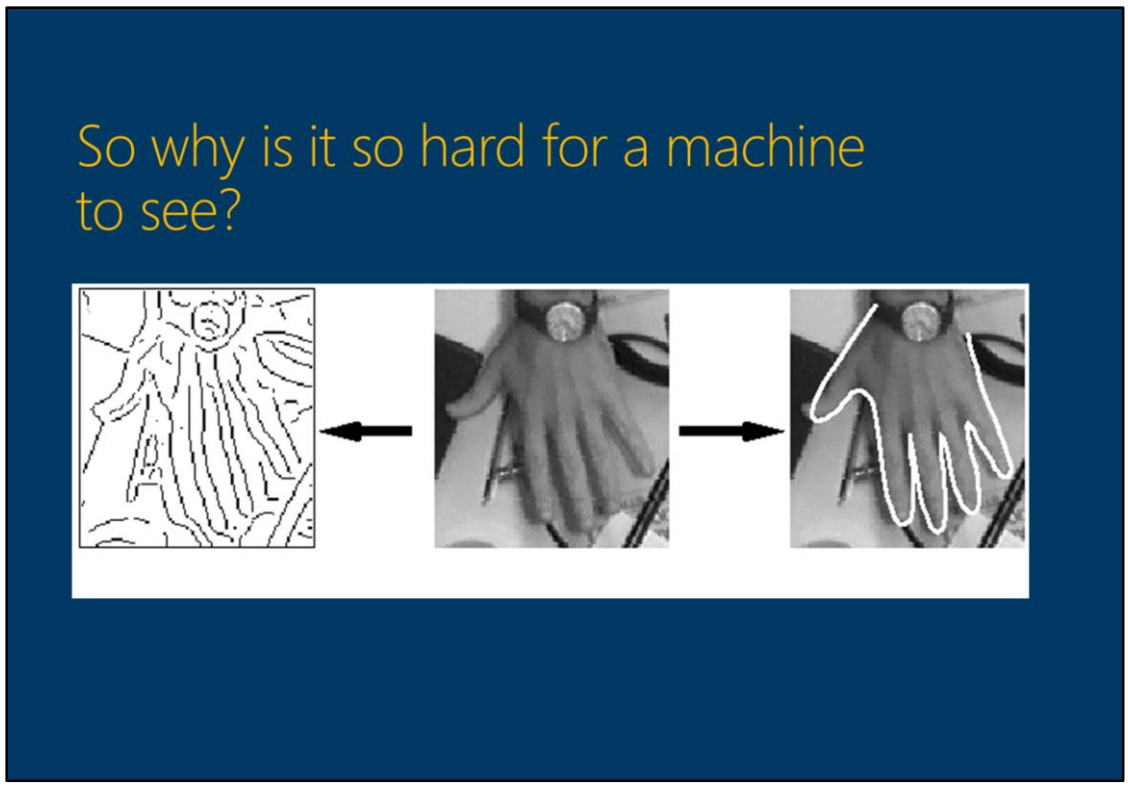


Three-dimensional vision is important, and a number of companies have marketed stereo vision systems that are either passive or active (generate their own illumination) and use either multiple optical devices together with triangulation, or active illumination and time of flight, to detect three-dimensional features. The camera illustrated is the first consumer 3D camera, the Microsoft Kinect, of which around 30 million have now sold. It also uses machine vision to detect the movement of human bodies, including the positions of all limbs, in real time.

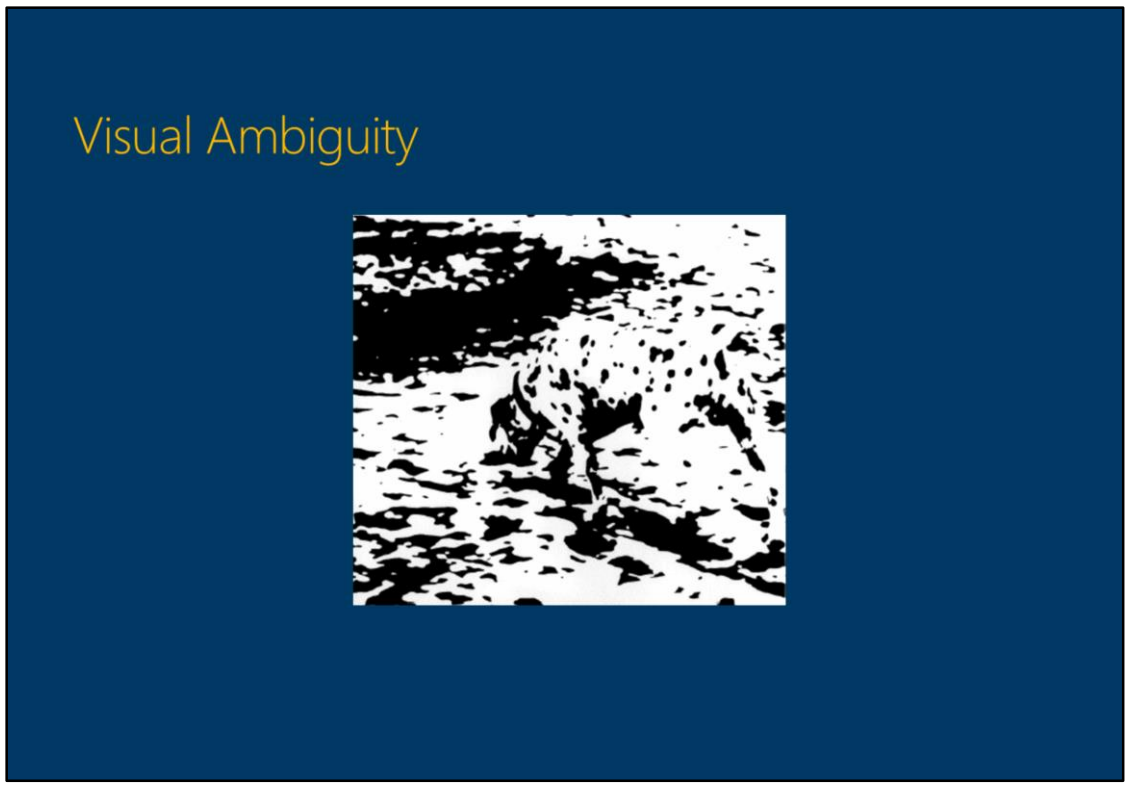


Machine vision has turned out to be much harder than expected. A summer project at MIT in 1966 planned to implement the basics of machine vision in the month of July. Some more challenging extensions would have to be left for August. Now, almost 50 years later, some thousands of researchers have achieved a good deal of progress, but there is still an awful lot to do before machine vision reaches the capability that humans and many animals enjoy.

(Sussman, Lamport & Guzman, 1966)



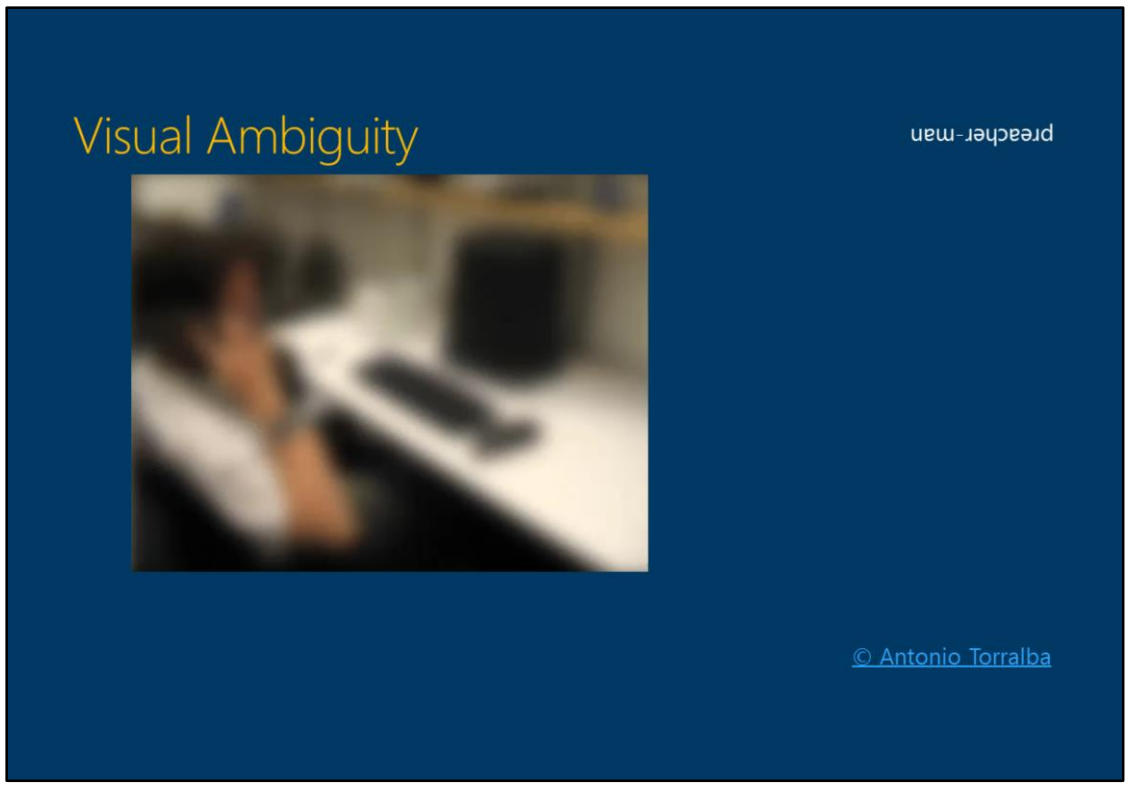
This figure illustrates something of what it is that makes vision so hard. Suppose the task is to locate precisely where the hand is in the image (centre). The result on the right would be good, and seems quite achievable to us as we inspect the image and, without any conscious effort, see the outline. But there is a very powerful computing device locate in the cranium of a human that is able to bring to bear very substantial computation, notwithstanding our lack of awareness that the computation is going on. On the left is the typical output of signal processing algorithms that aim to bring out the outline of the hand as a contour of high contrast. Sure enough the outline is there, though not quite unbroken. But it is confounded by numerous additional contours corresponding to clutter in the background and shadows and markings in the foreground. The clutter, shadows and markings, dealt with effortlessly by our brains and screened out from our attention, have to be dealt with explicitly by machine vision algorithms.



A further illustration of the nature of the machine vision challenge. Unless you have seen it before (and it is a staple of psychology textbooks) you may not recognise anything. But there is a prompt at the top right of the next page. Did that help? Many of you will now see the object, and what is more, can never now return to the state of ignorance in which you began.

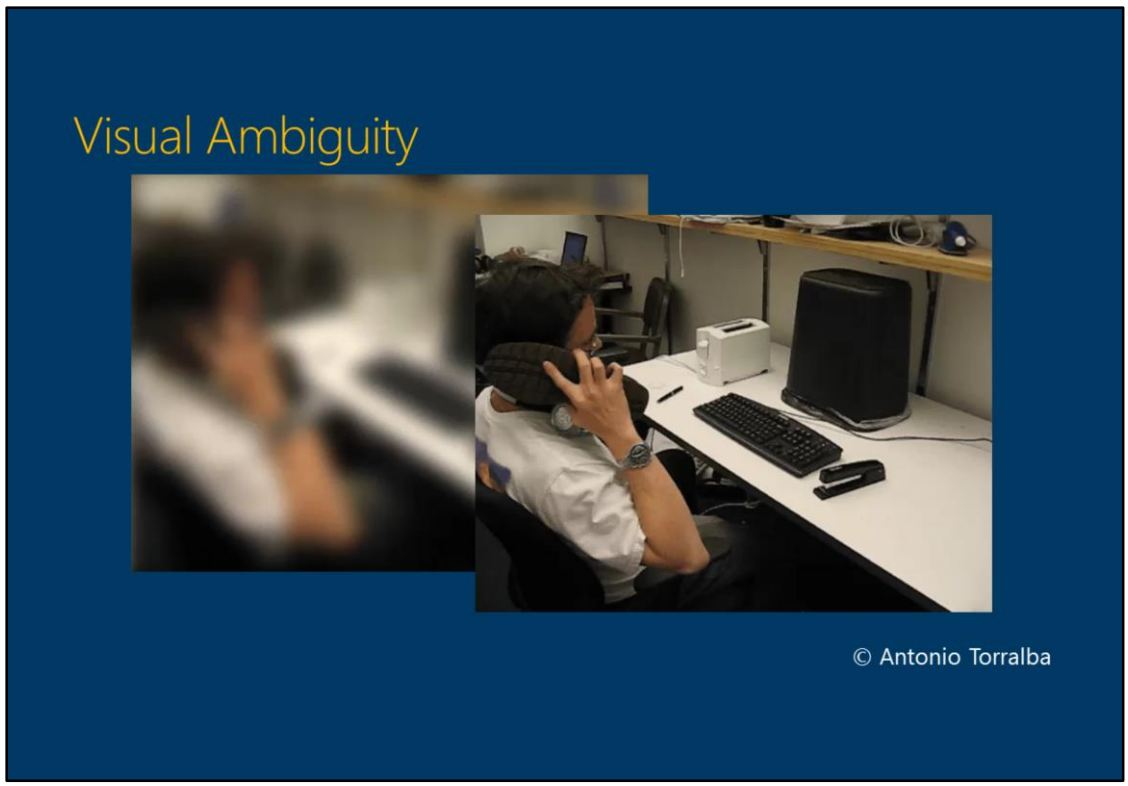
If the last one was too easy, or too hard, try this one. If you can't see the object right away, then the thumbnail on the next slide may help.

If the thumbnail still doesn't help, then look at the verbal prompt at the top right of the next page.



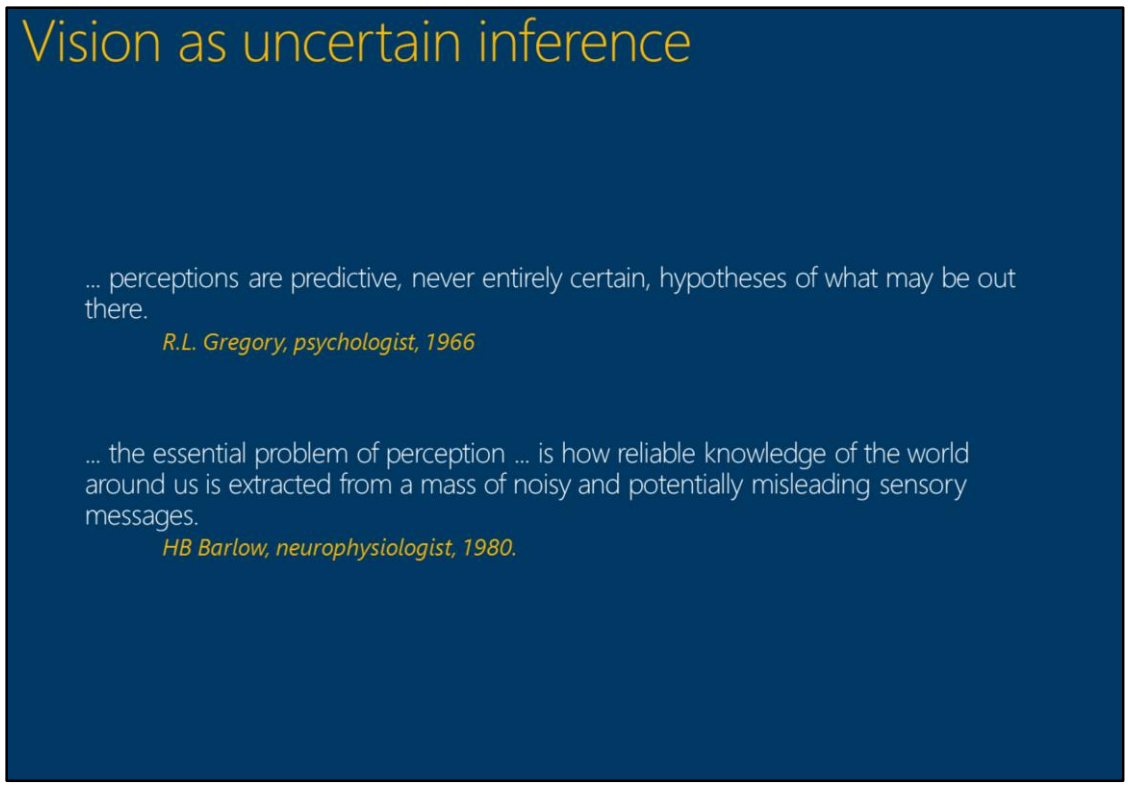
One last playful example before real work begins. Here is a blurry image of a man using a telephone in front of his keyboard and mouse.

© Antonio Torralba



But it turns out the telephone is in fact a shoe and the mouse is a stapler. It seems that, as humans doing vision, we are happy to build hypotheses to explain such visual evidence as is available – and then to revise the hypotheses as better evidence appears.

The previous examples – the dalmation and the preacher – also illustrate the ability of human vision to survive extremes of visual noise and ambiguity, and nonetheless find evidence for hypotheses about the content of the scenes.



The celebrated psychologist Richard Gregory has consistently expressed this view of what vision is doing – testing hypotheses against such evidence as may be available in a visual scene. Similarly, the conclusion of the neurophysiologist Horace Barlow, after many years of study of visual physiology, is that perception is about extracting knowledge about the visible world in the presence of noise and ambiguity.

So this also is what machine vision has to do, and algorithms need to be designed to do it. The algorithms must have an inherent capability to deal with noise and uncertainty. Probability is the calculus of uncertainty, and so we should expect the ideas of probability to be important in the design of those algorithms.

Computation steps for a seeing machine

- Segmentation
  - mark foreground objects
- Classification
  - label pedestrians
- Tracking
  - pedestrians in motion
- Control
  - steering and braking

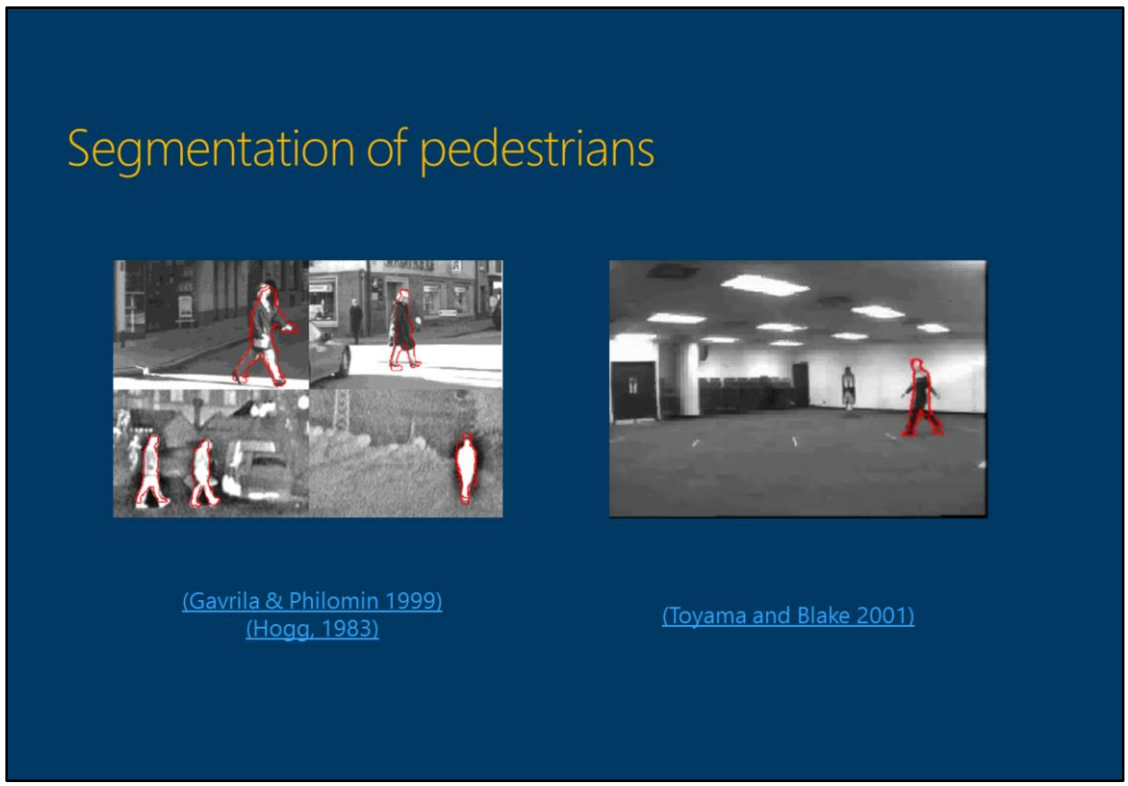
Mercedes -- visual safety systems on E-class and S-class 2013

(Keller, Enzweiler & Gavrila, 2011)

Cars with autonomous visual capability are already on the market. The system illustrated here from Mercedes is able to brake when a pedestrian steps out in front of the car, avoiding collision entirely in some cases, and mitigating injury in many others. The design of the vision system inside the cars involves four main steps. First is *segmentation*, in which objects which may be pedestrians are delineated from the background. Next is classification in which the identity of an object as a pedestrian is verified. The pedestrians are likely to be moving, so their motion must be *tracked*. Lastly the tracked motion is used in the car's actuation systems to control braking.

This lecture will address principally the first of these steps, the segmentation.

(Keller, Enzweiler & Gavrila, 2011)

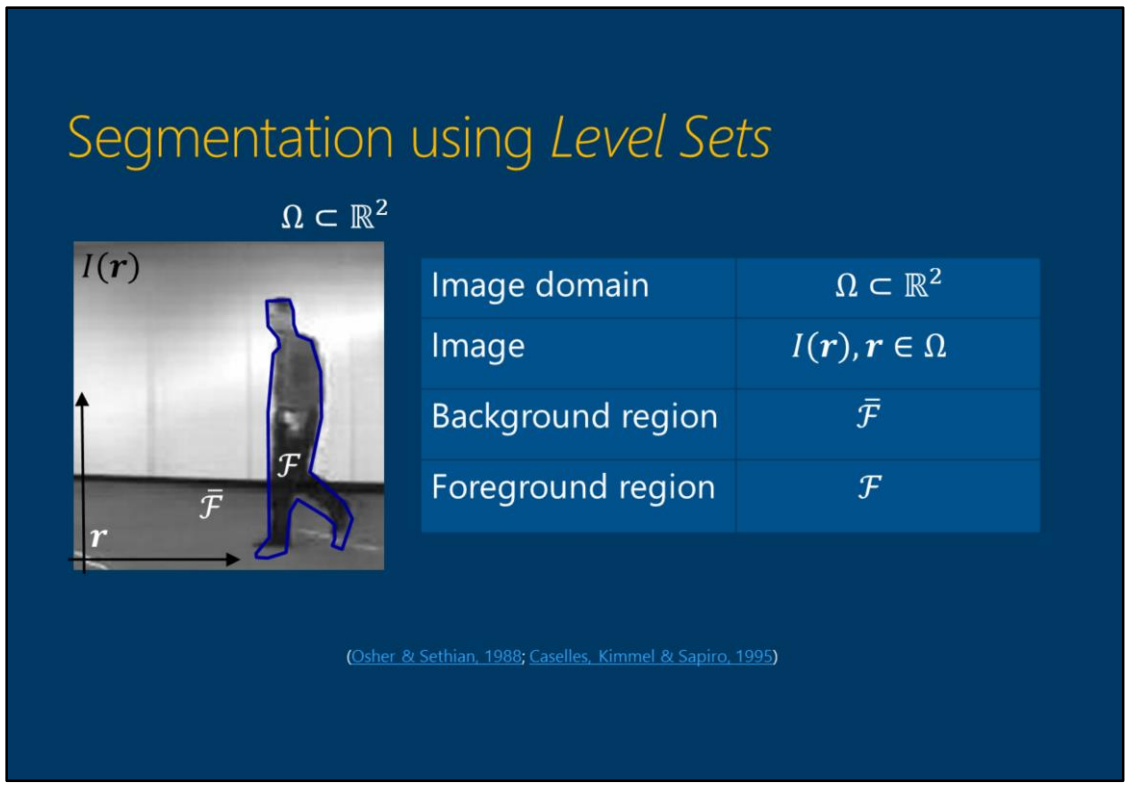


The task of segmentation requires foreground objects – in this case pedestrians – to be explicitly delineated in the scene. This has to be done not only on individual snapshots, but taking account of continuous motion, both of the object and of the (car-mounted) camera.

(Gavrila & Philomin 1999)

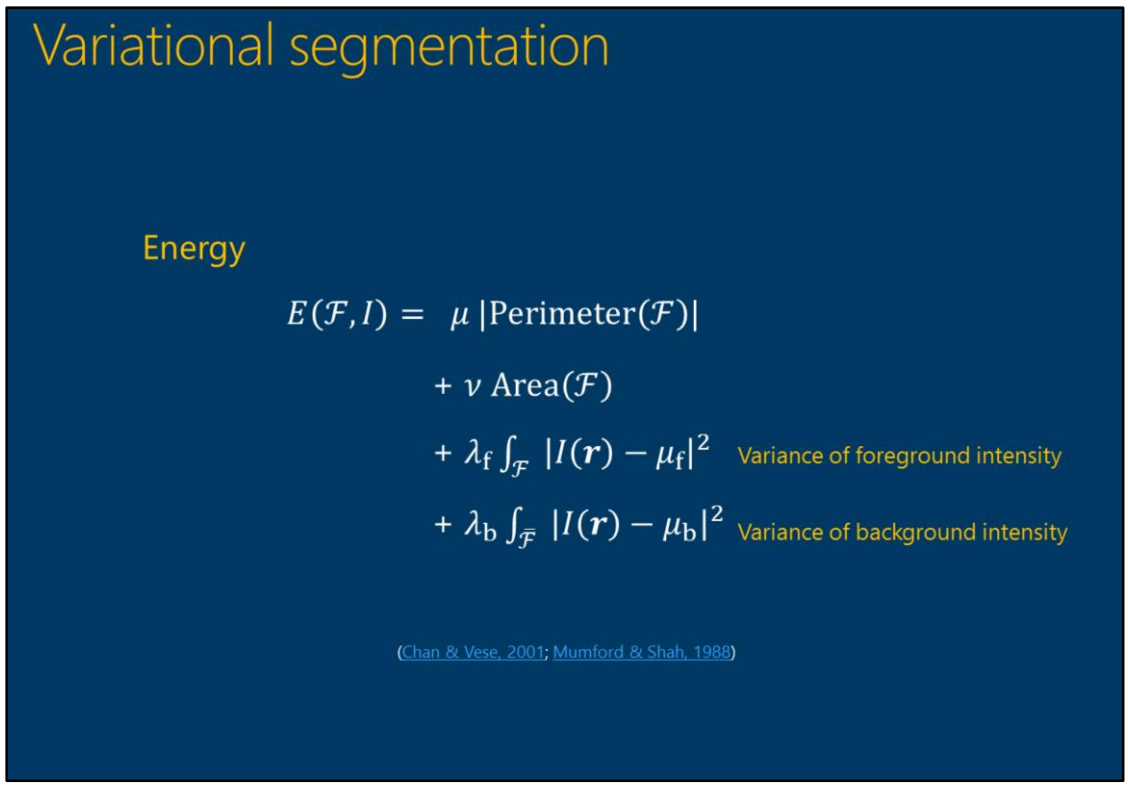
(Hogg, 1983)

(Toyama and Blake 2001)



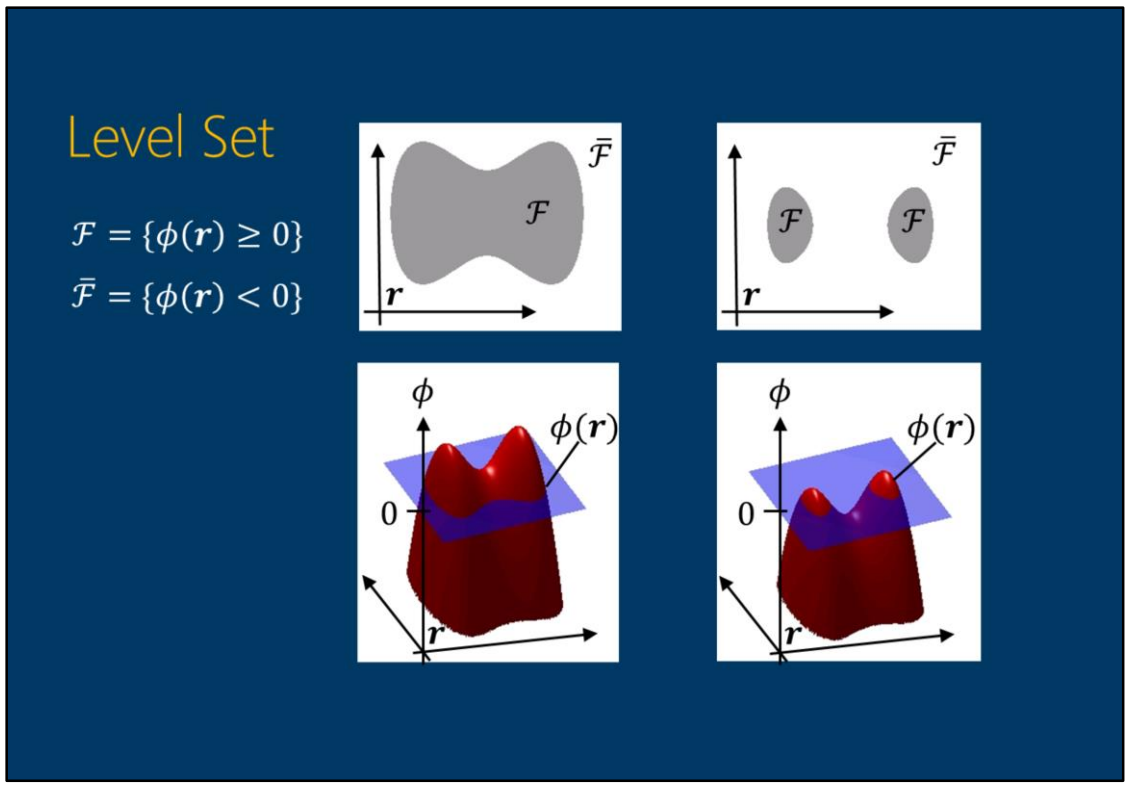
Here is the setting and mathematical notation for thinking about the segmentation problem.

(Osher & Sethian, 1988; Caselles, Kimmel & Sapiro, 1995)

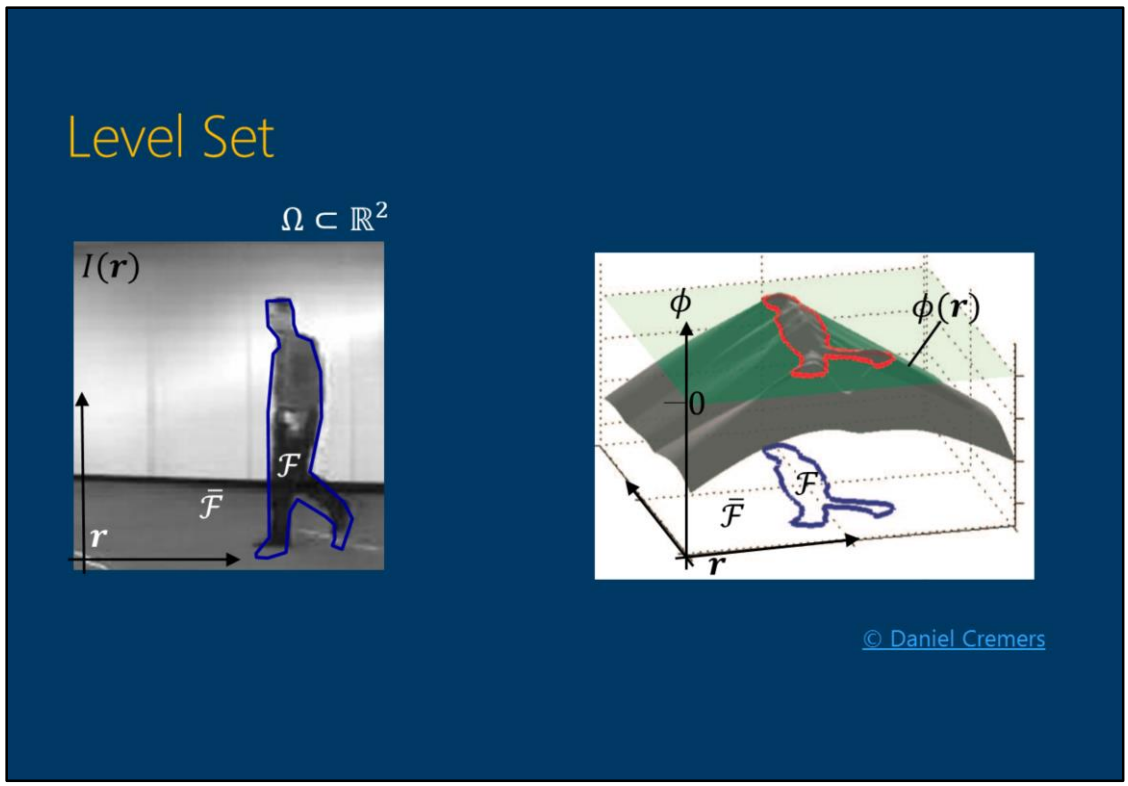


The aim is to minimize the energy E with respect to the variable foreground set $\mathcal{F}$, and in principle also with respect to the unknown mean intensity values for foreground and background, but we neglect them for the time being, assuming them to be known and fixed. However, to express the energy as a functional, amenable to variational calculus, the variable set needs to be replaced by some sort of function representing $\mathcal{F}$.

(Chan & Vese, 2001; Mumford & Shah, 1988)



A popular way to represent $\mathcal{F}$ as a function is by means of the "level set" function . This is by no means the only way to represent $\mathcal{F}$, but it has the great advantage of allowing variable topology. That is illustrated here where the set on the left consists of a single connected component. Then on the right, following a change in ϕ, the set $\mathcal{F}$ splits into 2 connected components.



Here is the level set construction illustrated for the original problem of pedestrian tracking. On the right, the level set function $\boldsymbol{\phi}$ is shown in grey, with the zero-level in green, and the zero-set as a red contour, whose pre-image is the blue boundary of $\boldsymbol{\mathcal{F}}$ in the image domain $\boldsymbol{\Omega}$.

Variational segmentation

Energy functional

$$E(\mathcal{F}, I) = \mu\,|\text{Perimeter}(\mathcal{F})| + \nu\,\text{Area}(\mathcal{F}) + \lambda_{\text{f}} \int_{\mathcal{F}} |I(\boldsymbol{r}) - \mu_{\text{f}}|^2 + \lambda_{\text{b}} \int_{\bar{\mathcal{F}}} |I(\boldsymbol{r}) - \mu_{\text{b}}|^2$$

Variance of foreground intensity

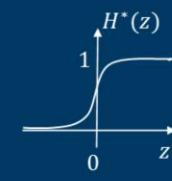
Variance of background intensity

Now express in terms of ϕ :

So now the job is to express the energy as a functional, in terms of the level set ϕ.

...Segmenting regions – level sets

Using ϕ and (smooth approx to) Heaviside step fn:



$$E(\mathcal{F}, I) = \cancel{\mu\,|\text{Perimeter}(\mathcal{F})|} \quad \mu \int_\Omega |\nabla H^*(\phi(\boldsymbol{r}))|$$

$$\cancel{+\,\nu\,\text{Area}(\mathcal{F})} \quad \nu \int_\Omega H^*(\phi(\boldsymbol{r}))$$

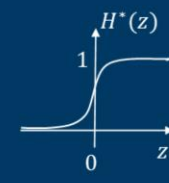
$$\cancel{+\,\lambda_\text{f} \int_\mathcal{F} |I(\boldsymbol{r}) - \mu_\text{f}|^2} \quad \lambda_\text{f} \int_\Omega H^*(\phi(\boldsymbol{r}))\, |I(\boldsymbol{r}) - \mu_\text{f}|^2$$

$$\cancel{+\,\lambda_\text{b} \int_\mathcal{F} |I(\boldsymbol{r}) - \mu_\text{b}|^2} \quad \lambda_\text{b} \int_\Omega \left(1 - H^*(\phi(\boldsymbol{r}))\right) |I(\boldsymbol{r}) - \mu_\text{f}|^2$$

Each term in E could be expressed as an integral over the image domain Ω, in terms of the Heavyside step function H. Furthermore, in place of H, a smooth approximation H^* is used to ensure that the dependence of E on ϕ is differentiable. If we had used H itself in the integrals above, then they would express $E(\mathcal{F}, I)$ *exactly*, but as it is they approximate E in a way that ensures differentiability.

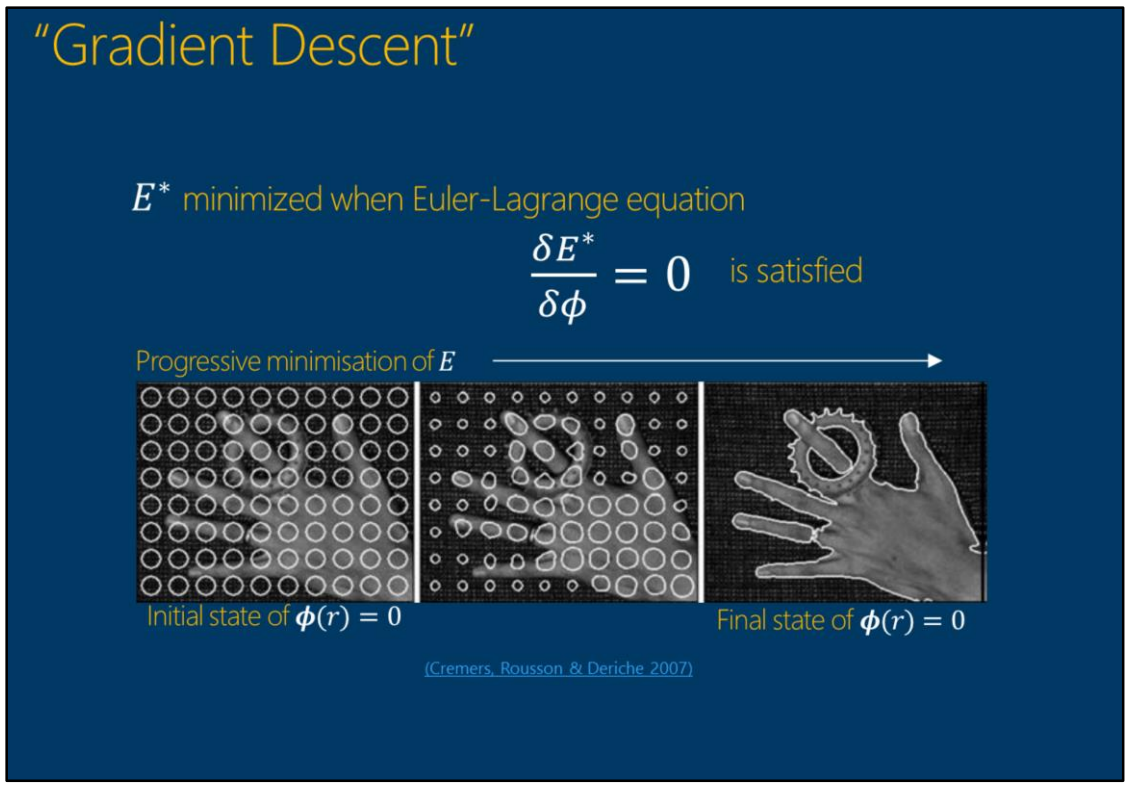
…Segmenting regions – level sets

Using ϕ and (smooth approx to) Heaviside step fn:



$$E^*(\phi, I) = \mu \int_\Omega |\nabla H^*(\phi(\boldsymbol{r}))|$$
$$+ \nu \int_\Omega H^*(\phi(\boldsymbol{r}))$$
$$+ \lambda_{\mathrm{f}} \int_\Omega H^*(\phi(\boldsymbol{r}))\, |I(\boldsymbol{r}) - \mu_{\mathrm{f}}|^2$$
$$+ \lambda_{\mathrm{b}} \int_\Omega \left(1 - H^*(\phi(\boldsymbol{r}))\right) |I(\boldsymbol{r}) - \mu_{\mathrm{b}}|^2$$

So here is the approximation to $\boldsymbol{E}$, a functional that we denote $\boldsymbol{E}^*$.



This illustration shows the results of an algorithm executing gradient descent on E^*, towards the minimum. The initial value of ϕ has a level set consisting of many circles, as shown, and the gradient descent moves progressively towards the outline of the object.

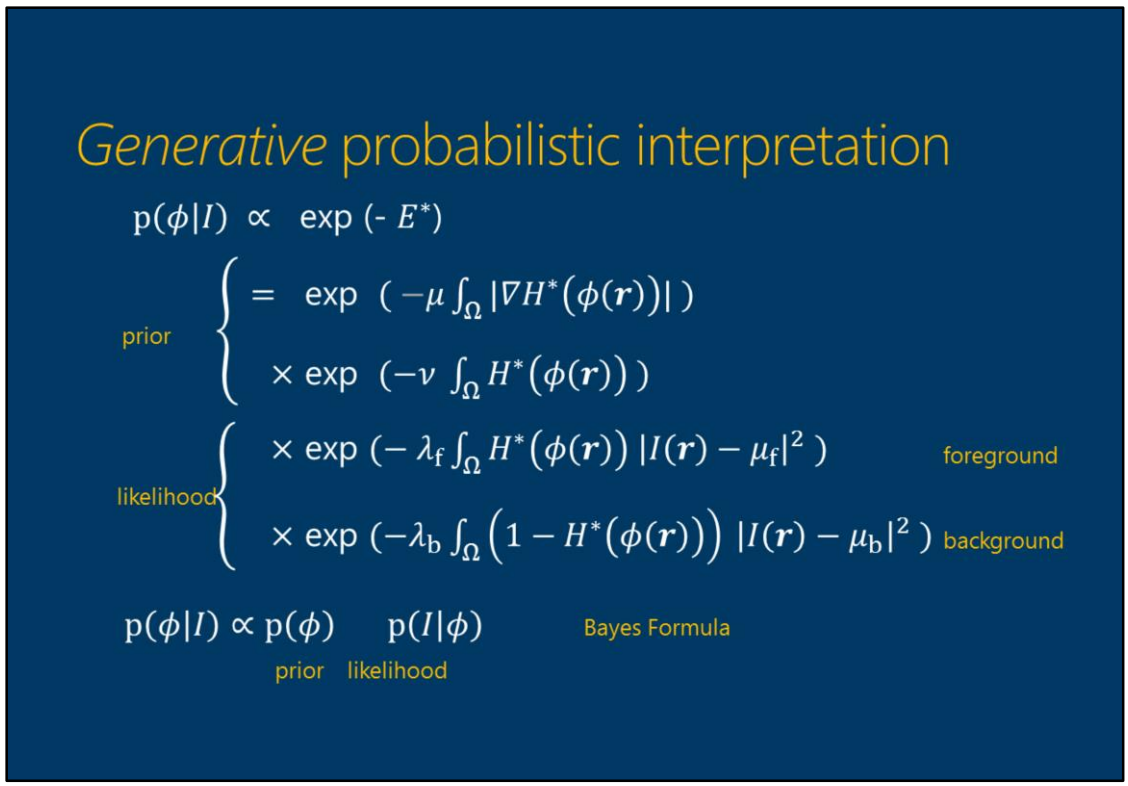
(Cremers, Rousson & Deriche 2007)

From energy → probability

energy $$E^*(\phi, I) = \mu \int_\Omega |\nabla H^*(\phi(\boldsymbol{r}))| + \nu \int_\Omega H^*(\phi(\boldsymbol{r})) + \lambda_{\mathrm{f}} \int_\Omega H^*(\phi(\boldsymbol{r}))\, |I(\boldsymbol{r}) - \mu_{\mathrm{f}}|^2 + \lambda_{\mathrm{b}} \int_\Omega \left(1 - H^*(\phi(\boldsymbol{r}))\right) |I(\boldsymbol{r}) - \mu_{\mathrm{b}}|^2$$

probability $\mathrm{p}(\phi|I) \propto \exp(-E^*)$

One of the main points of this lecture is that probability is a necessary part of the shape tracking framework, because of the need to deal effectively with the noise and ambiguity that is inherent to the vision task. So far we have seen how variational methods address the noise problem to some degree. But the most powerful methods for dealing with noise and ambiguity exploit specific properties of the class of objects. This can be done by using a posterior probability distribution instead of the Energy functional. The first step towards seeing how this works, is to transform the functional E^* to a posterior distribution by exponentiating as above – what could be more appropriate for the *Gibbs* lecture?



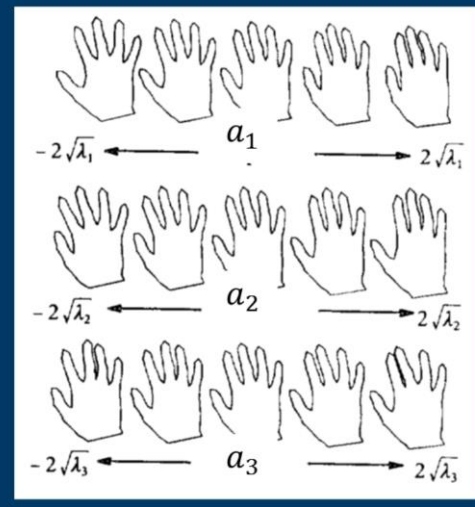
Here is the probability distribution written out term by term. The four additive terms from the functional $\boldsymbol{E}$ become four multiplicative factors as shown. Comparing this distribution with the Bayes formula for a posterior distribution, it is evident that the first two terms are independent of the data $\boldsymbol{I}$ – they correspond to the prior distribution. The second two terms are data dependent, and they correspond to the data likelihood.

We refer to the posterior distribution here as a *generative* model because it constitutes a full explanation of the data $\boldsymbol{I(r)}$ in terms of the hypothesised shape $\boldsymbol{\phi}$.

So now the opportunity is to replace each of the prior and likelihood terms by terms which relate specifically to the object class. The prior distribution can capture the properties of shape for the object class. The likelihood can capture the properties of texture.

Prior shape model

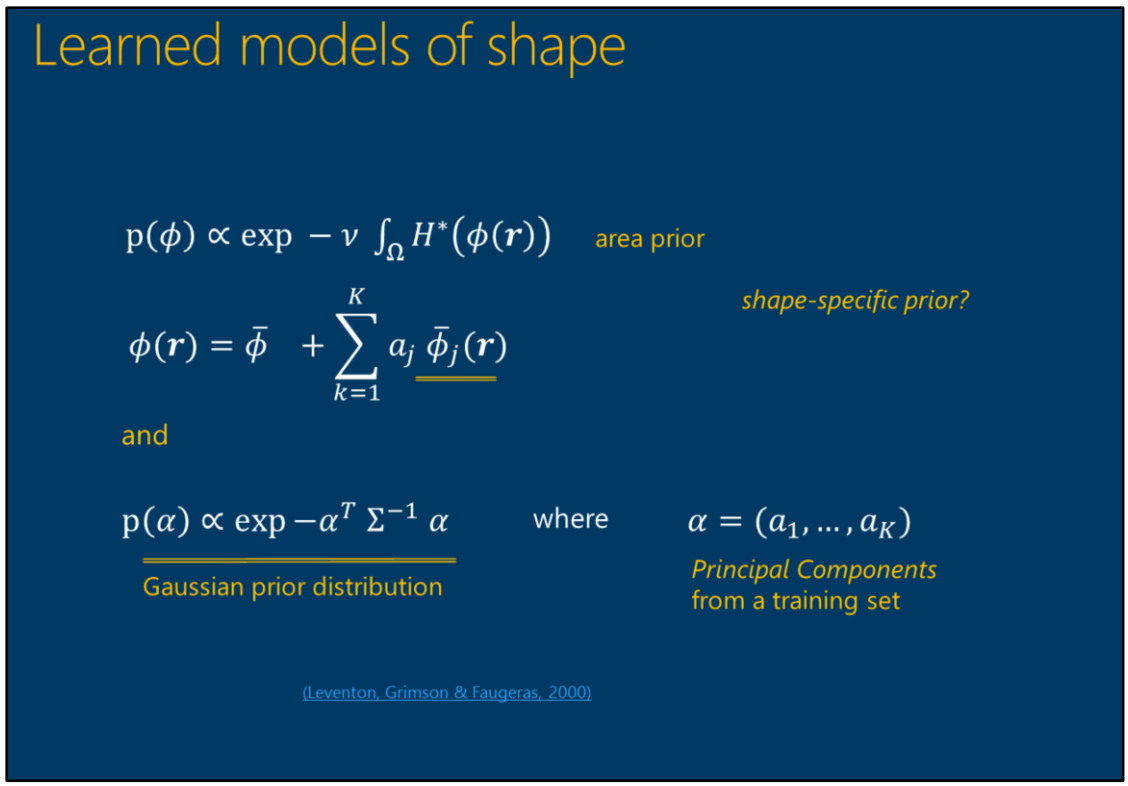
– Principal Components



(Cootes and Taylor, 1995)

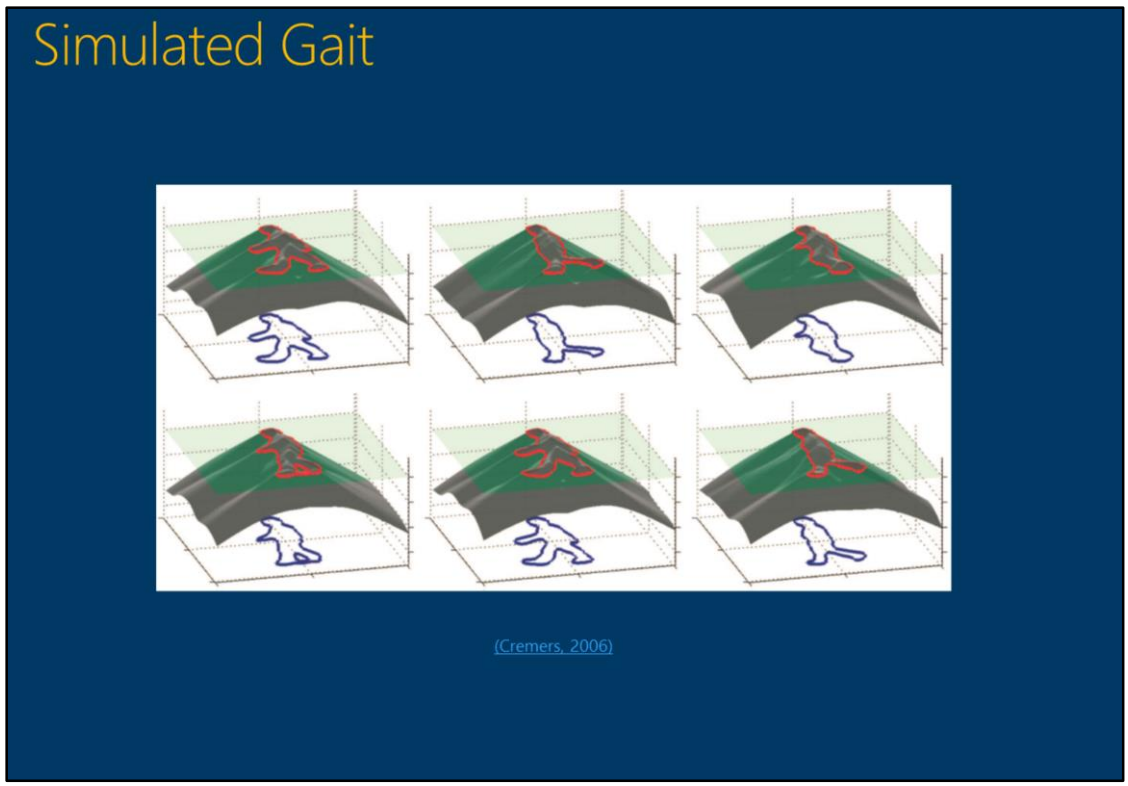
One way of acquiring a prior model for shape is to learn it from examples. Here, a large training set of hand outlines are summarised using the standard statistical technique of Principal Components Analysis (PCA). The result is a low-dimensional space that captures almost all the variability of the training set. For instance the first dimension a_1 of this space captures the opening and closing of the fingers. The second dimension emphasises thumb adduction, and so forth. Just a few dimensions, of the order of a dozen, are sufficient to capture all most all of the shape variation. The low dimensional $a_1, a_2, \ldots$ is now a representation of shape, restricted to a subset of shapes similar to those in the training set. What is more, PCA also yields a Gaussian probability distribution over the $a_1, a_2, \ldots$, which is the prior distribution over the shape subset.

(Cootes and Taylor, 1995)



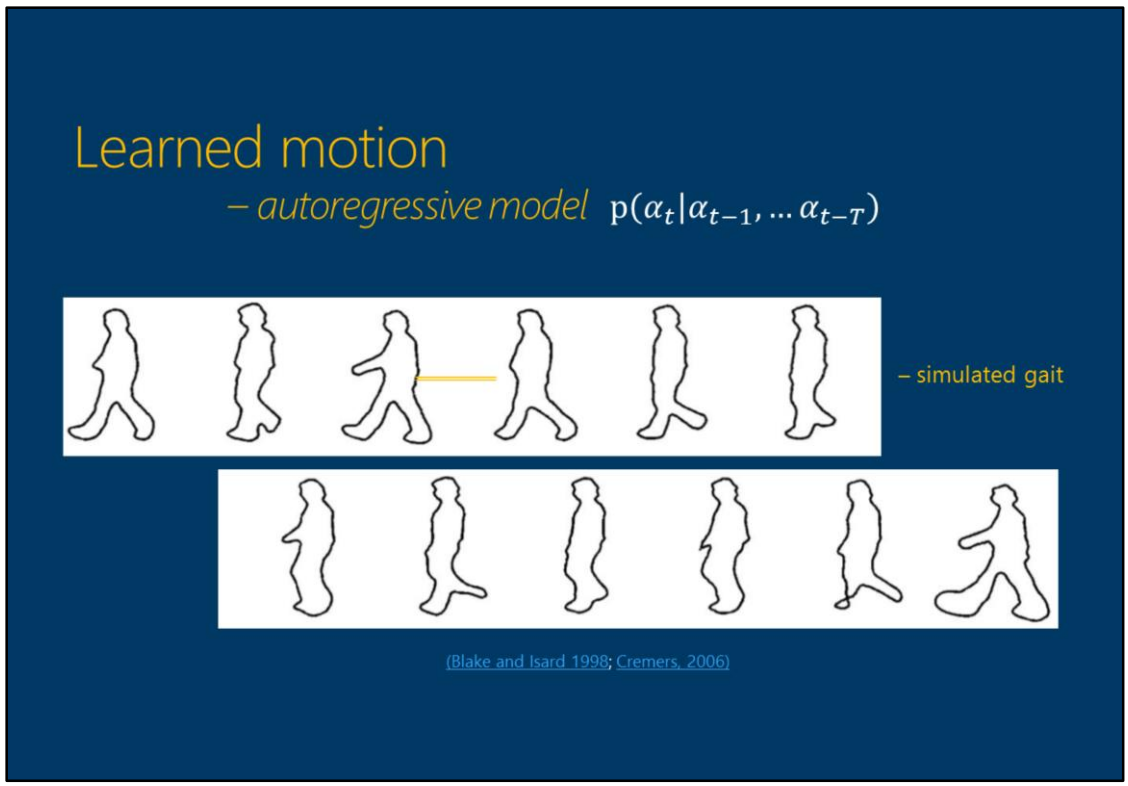
The PCA methodology can be applied successfully to shape represented by ϕ. Training shapes are represented by their level set functions, which are summarised to give a low dimensional representation of allowed shapes ϕ in terms of α, together with the Gaussian prior distribution for shapes over α as shown.

(Leventon, Grimson & Faugeras, 2000)



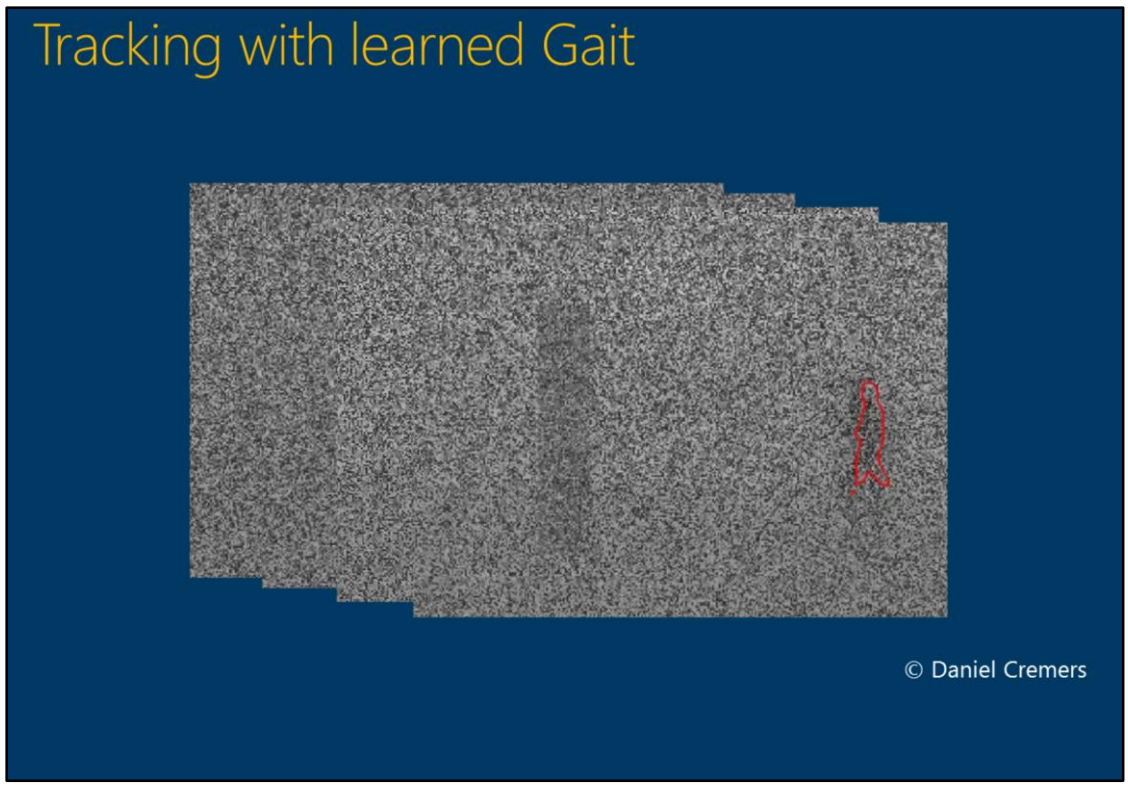
Here is a typical set of outline shapes for a walking person, represented by their level-set functions. (Note the level set function for a given shape is not unique, so the *signed distance function*, suitably smoothed, is used to construct the level-set function.) See how, in the bottom left frame, the topology of the outline changes, something that is handled happily by the level-set formulation, and which would defeat other representations of shape such as parametric outline curves.

(Cremers, 2006)



A further elaboration of prior shape modelling acknowledges that the shape evolves over time, and that the choice of shape, given the history of shapes, is strongly conditioned by the immediate history. Hence the prior distribution for shape at time *t* is modelled by a Markov model as shown, for example an autoregressive model.

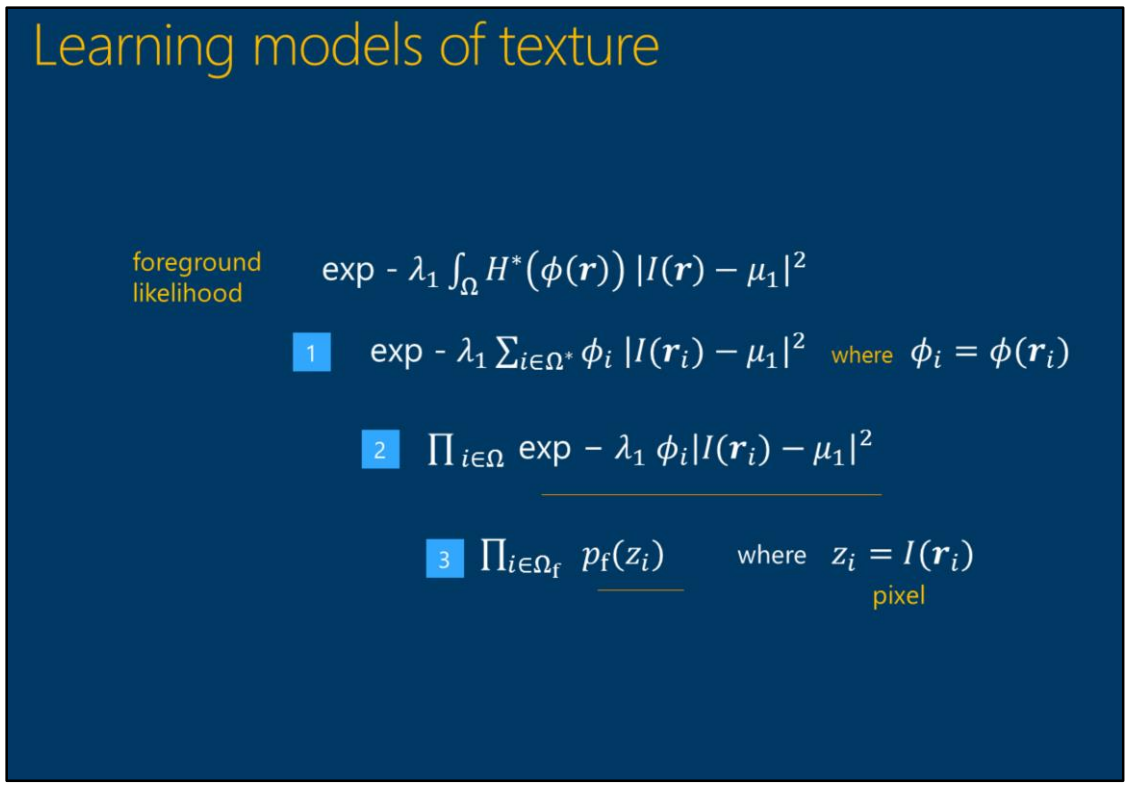
(Blake and Isard 1998 Cremers, 2006)



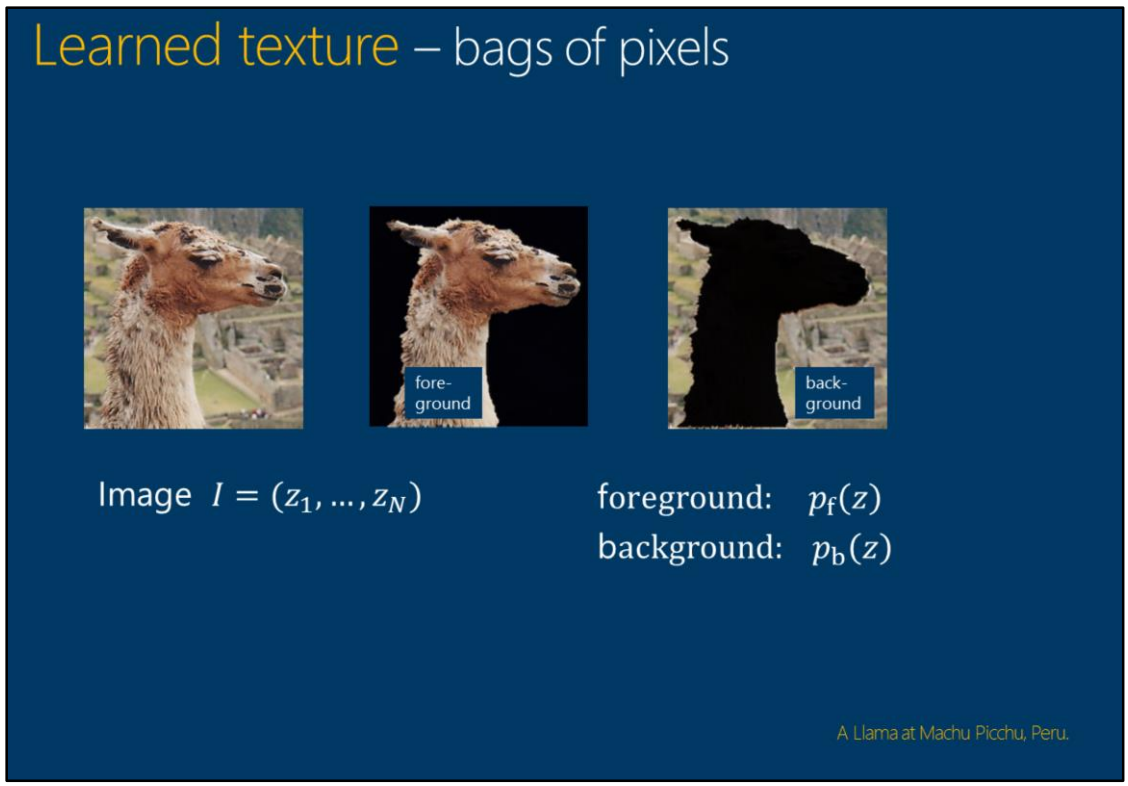
Tracking with learned dynamical model , in presence of noise and occlusion.

1. Noisy walker
2. Tracked with a shape model but no translation prior
3. The same model but now with an occluder – tracking fails
4. Combined translation/deformation prior, now succeeds even with the occluder

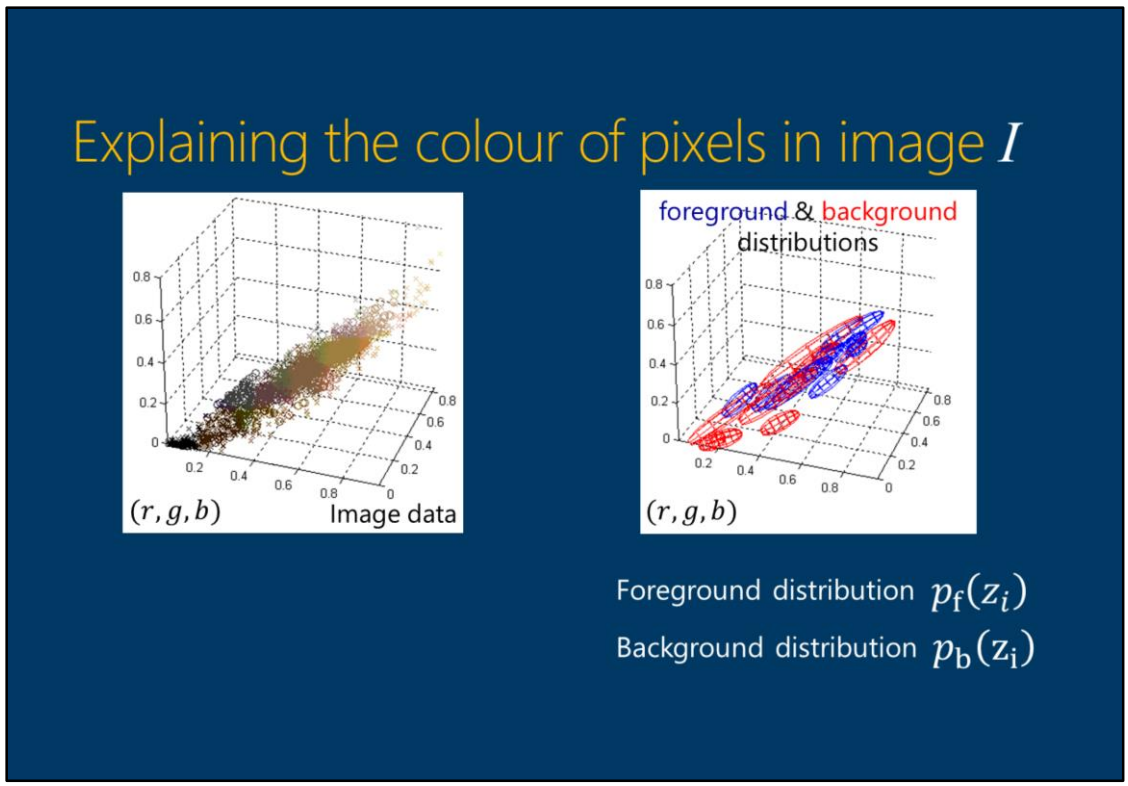
These rather impressive demos of the power of prior models are videos, unfortunately not currently available in this canned version of the talk. Maybe later.



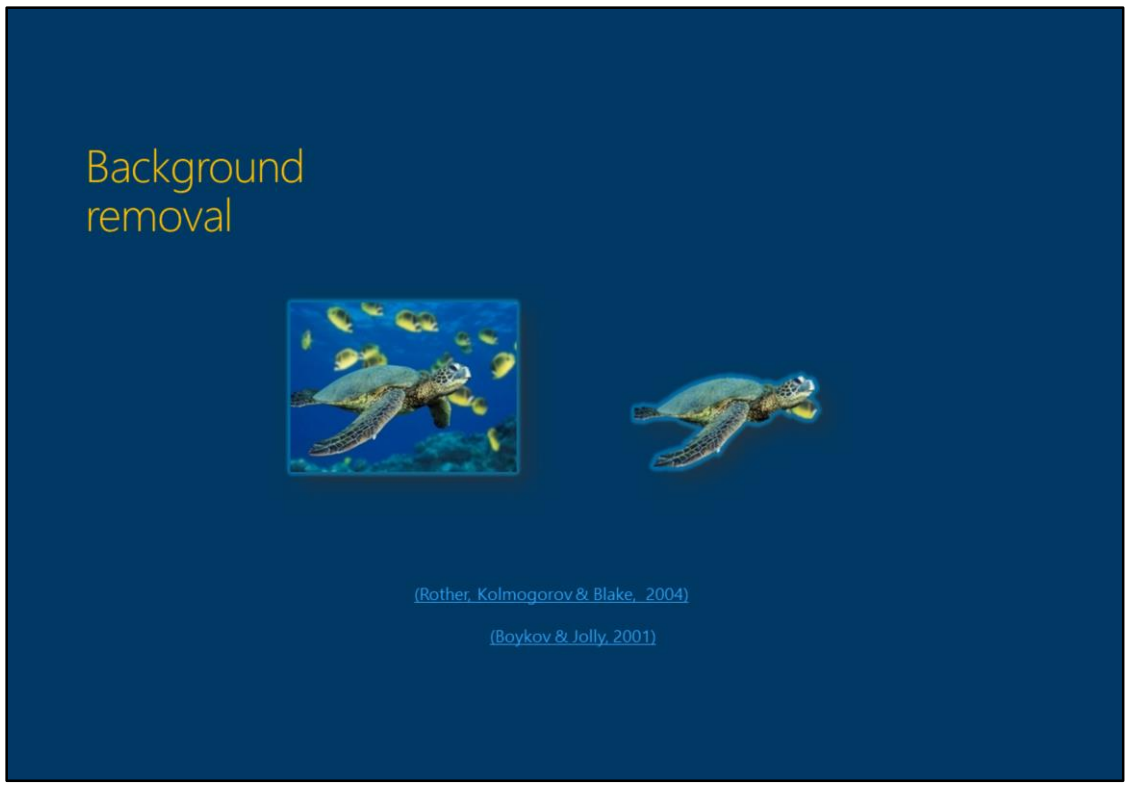
It has been shown how a class-specific prior model of shape can be built by learning from examples. Now for a specific model of object texture. Starting with the foreground likelihood term earlier, it is re-expressed in a new form, in three steps. First the integral is replaced by a sum over image pixels. Second, the sum is taken outside the exponential and becomes a product over pixels. Third, the exponential expression is expressed more generally as a general distribution p_f over pixel colour, and this invites the learning of that distribution for foreground appearance.



A learned distribution like p_{f} is often referred to as a *bag of pixels* model. It regards all the pixels of the foreground (and similarly of the background) as being drawn independently from one foreground colour distribution. This model is over-simple, in that if you simulate from it you get nothing very like the image. Yet in practice it still captures enough of the superficial properties of the image to succeed at some kinds of foreground/background analysis. (It can also be shown that the resulting analysis is also precisely consistent with certain more elaborate models of appearance, such as a Markov model over pixels, which would give more plausible simulations.)



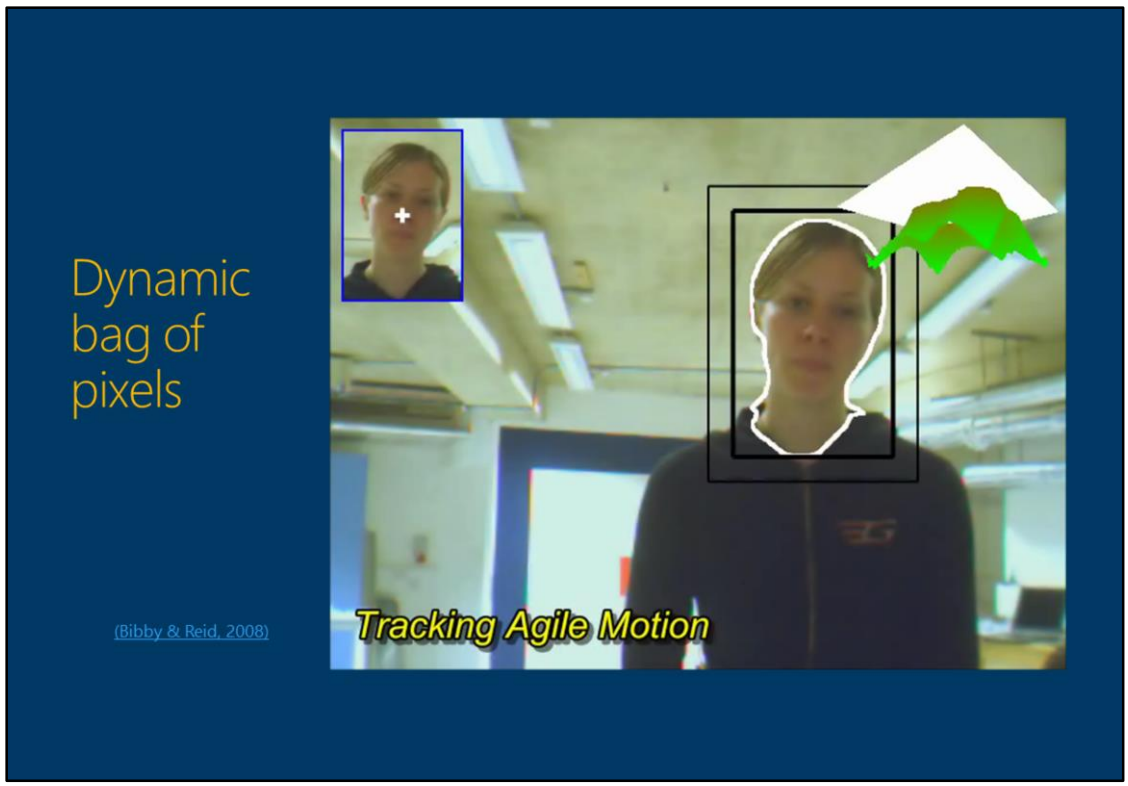
Here is an illustration of learned foreground and background for the Llama image. On the left, the colours of all pixels are shown mapped into RGB colour space. There is no very apparent separation between foreground and background distributions. This is because the Llama is somewhat camouflaged against its background. On the right, the foreground distribution (blue) is modelled as a set of Gaussian components, and similarly the background, and it is clear that there is some separation between foreground and background distributions, but not a complete separation. An attempt to distinguish between foreground and back ground purely on the basis of colour would fail, but succeeds when combined with the general shape prior from the variational energy $\boldsymbol{E}$ earlier.



The principles of the previous slides underlie software for background removal in interactive graphics that is widely used, for instance in all Microsoft Office applications since 2010.

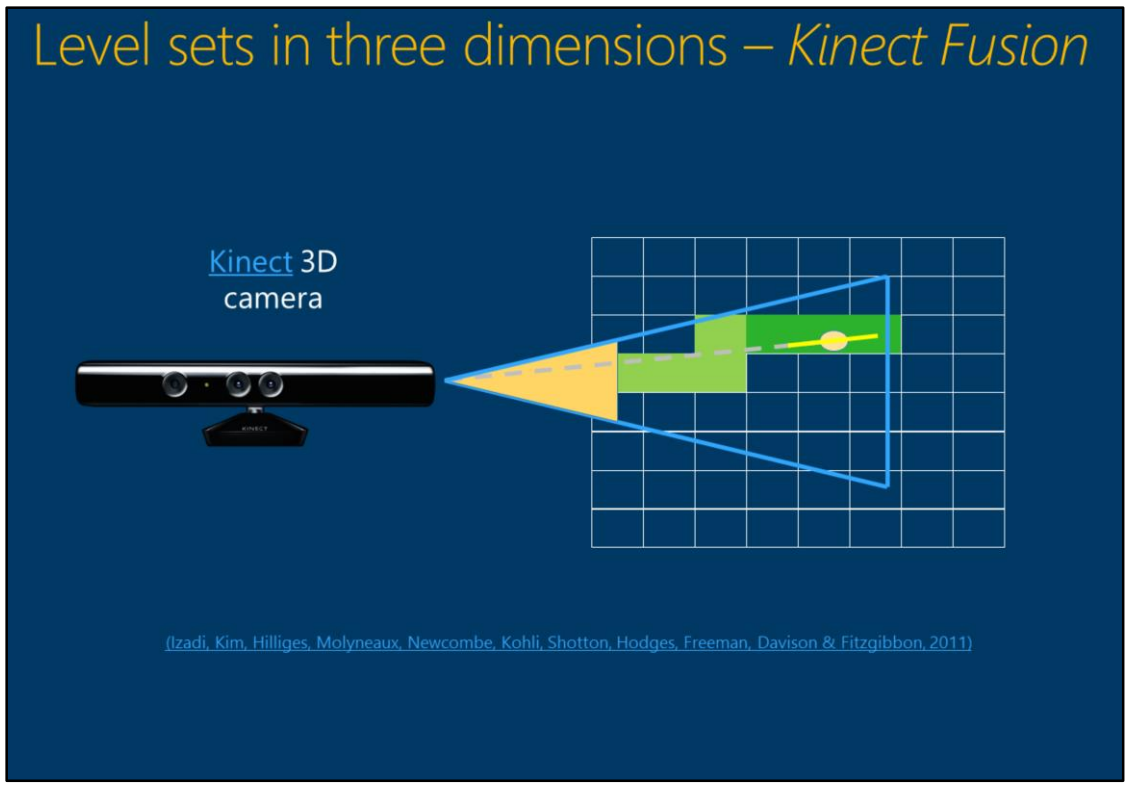
(Rother, Kolmogorov & Blake, 2004)

(Boykov & Jolly, 2001)



Research from Oxford shows how level-set representation of shape together with bag-of-pixel models can be used to segment very agile motion.

(Bibby & Reid, 2008)



In the next few slides, we digress temporarily from automatic segmentation to explore another capability of level-sets, this time in three dimensions. The Kinect 3D camera is capable of capturing 3D depth maps of a scene rapidly. The aim of *Kinect Fusion* is to aggregate overlapping depth maps obtained from a moving camera into one, high quality depth map, with the gaps that appear in a single depth-map filled in, and the measurement noise abated by statistical fusion. The medium for fusion of depth information is a level set function on a 3D grid. From an individual viewpoint, the level-set function is in fact a pixel-by-pixel map for the probability of occupancy. In multiple views, the individual maps are overlaid in a global coordinate frame and fused statistically to obtain a refined level-set function.

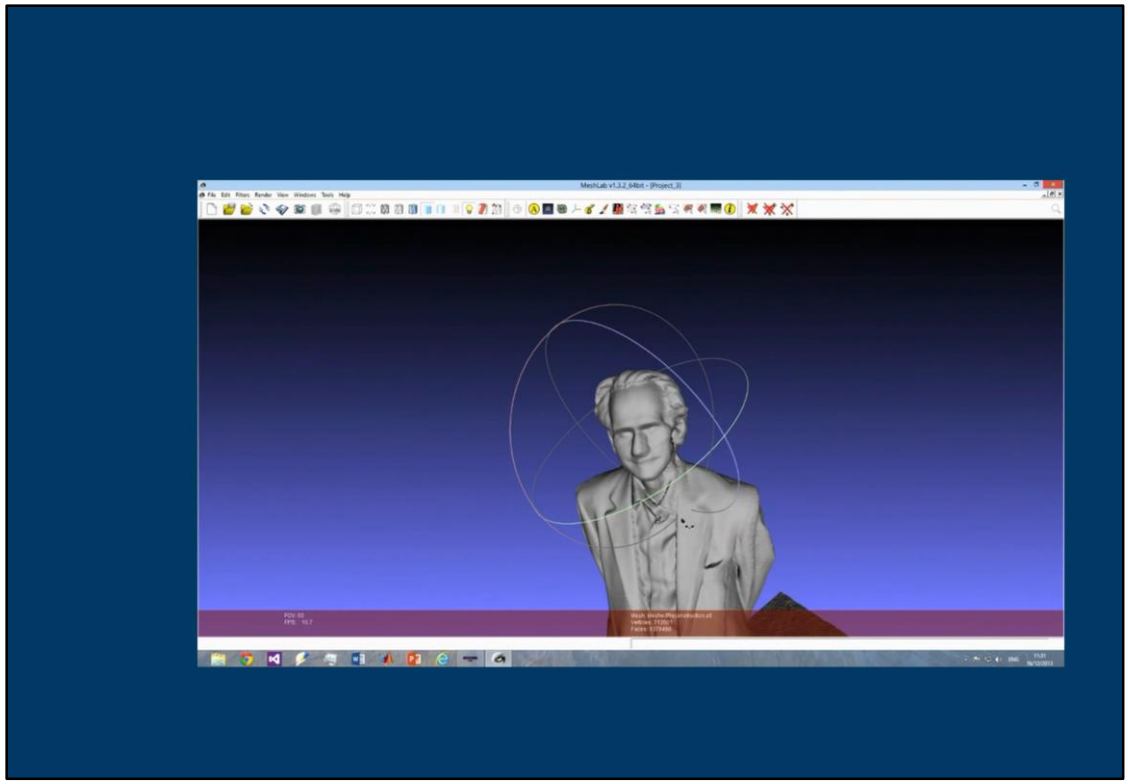
(Izadi, Kim, Hilliges, Molyneaux, Newcombe, Kohli, Shotton, Hodges, Freeman, Davison & Fitzgibbon, 2011)

Bill Gates at the University of Washington in 2013, demonstrating Kinect fusion.

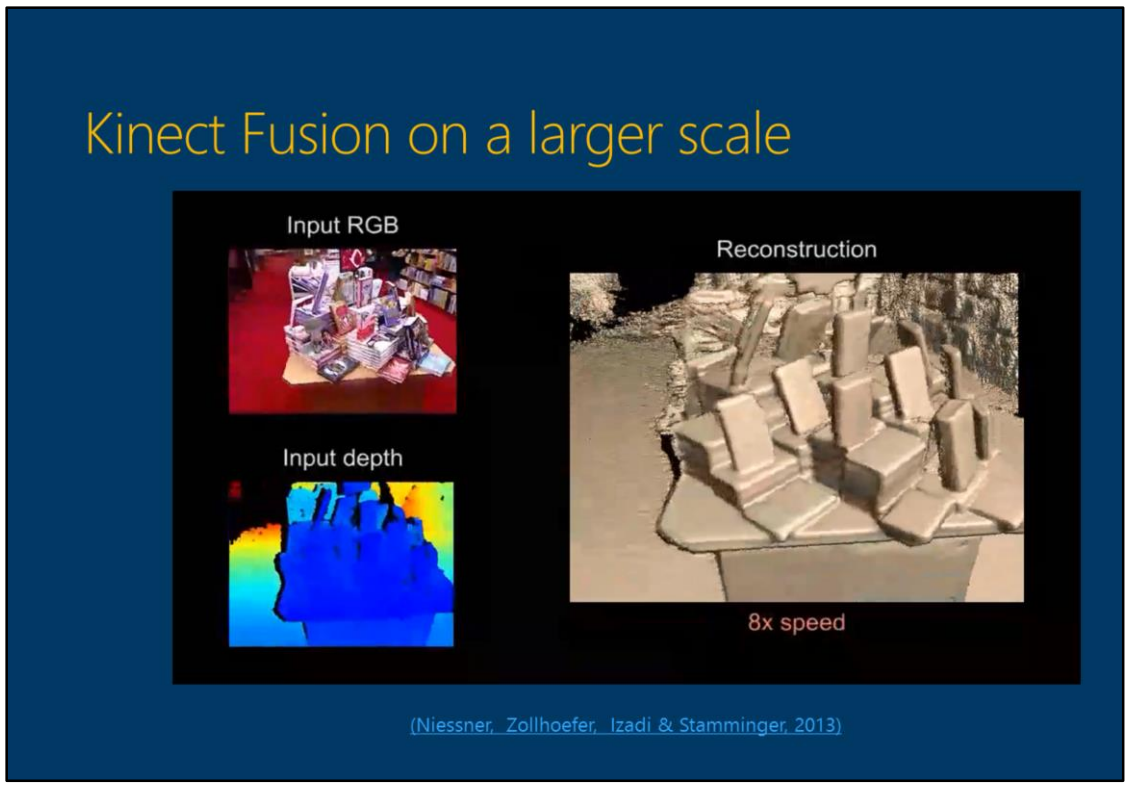
Gates is scanning a toy house by moving the Kinect camera around it. The top left shows the depth map from an individual viewpoint, and the top-centre shows the corresponding colour view, with a depth-map triangulation at the top right. On the bottom row is the depth map from fusing multiple views.

Kinect Fusion: scanning the death mask of Sir Isaac Newton, at the Royal Society

Kinect Fusion doesn't only work on dead mathematicians. Here the combinatorics expert Bela Bollobas is scanned also….

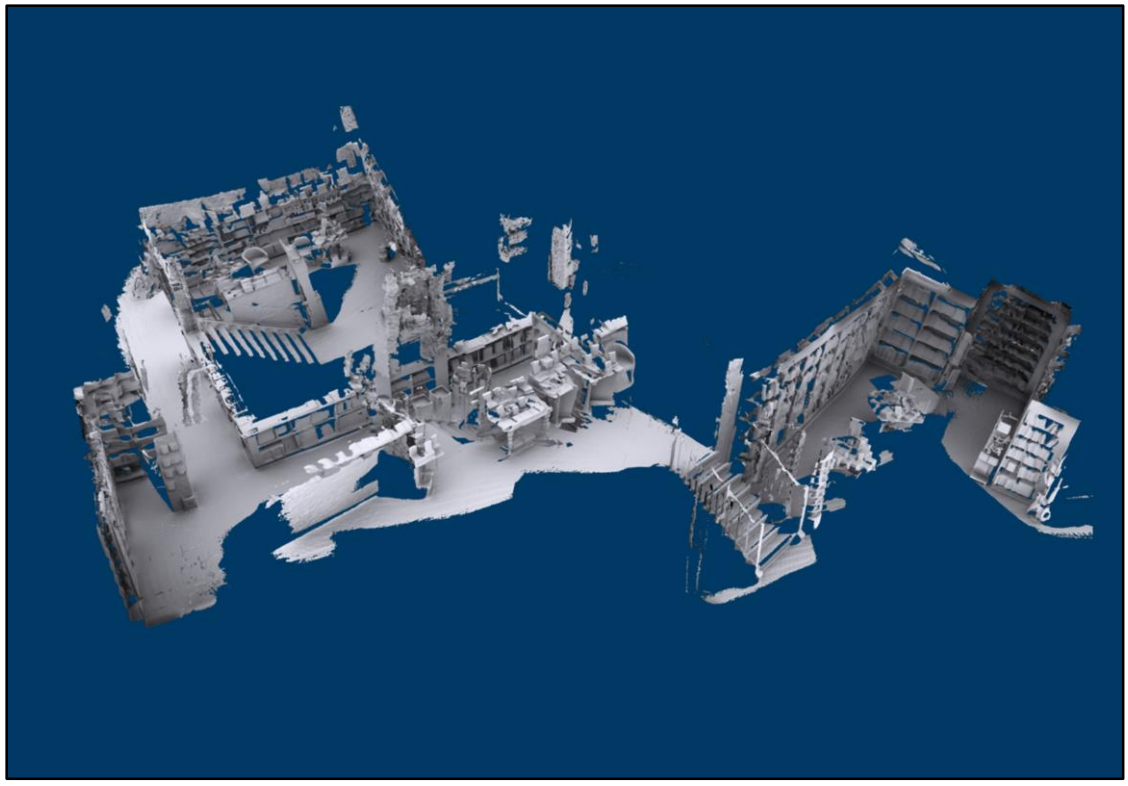


… and captured in 3 dimensions.

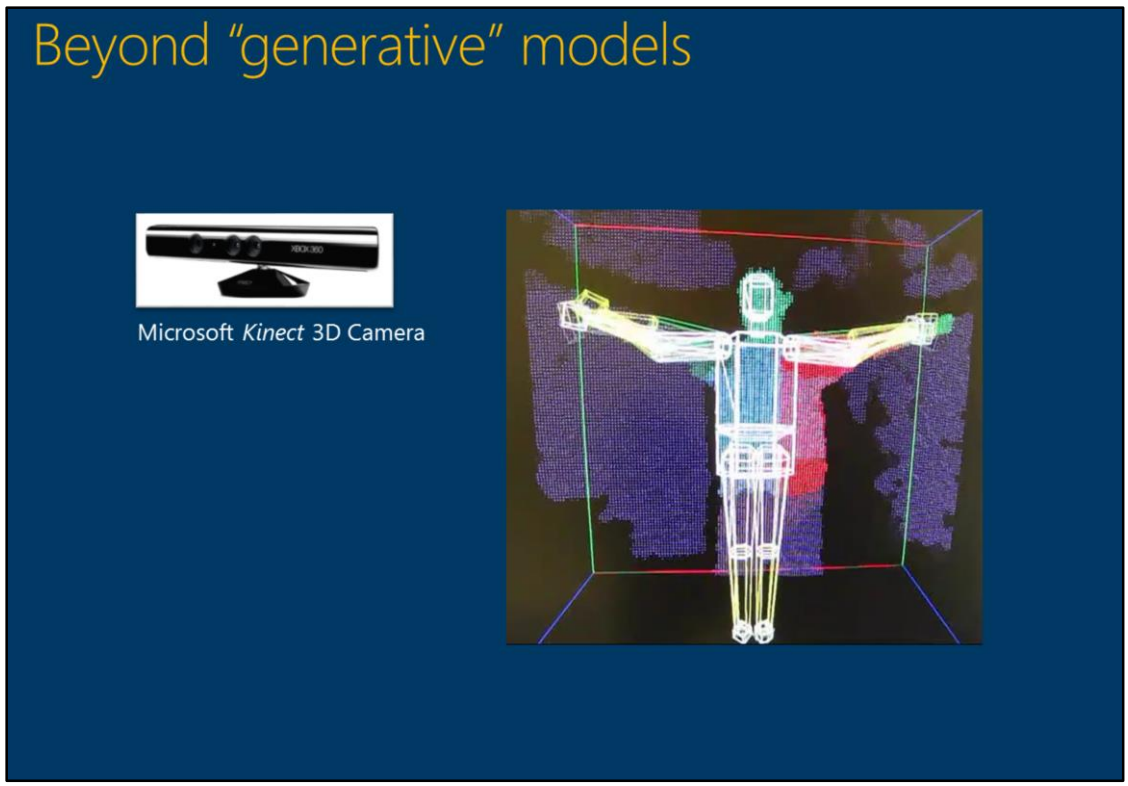


Finally a tour round a Cambridge bookshop, using a version of Kinect Fusion that is capable of extended operation without running out of memory or using excessive computation.

(Niessner, Zollhoefer, Izadi & Stamminger, 2013)



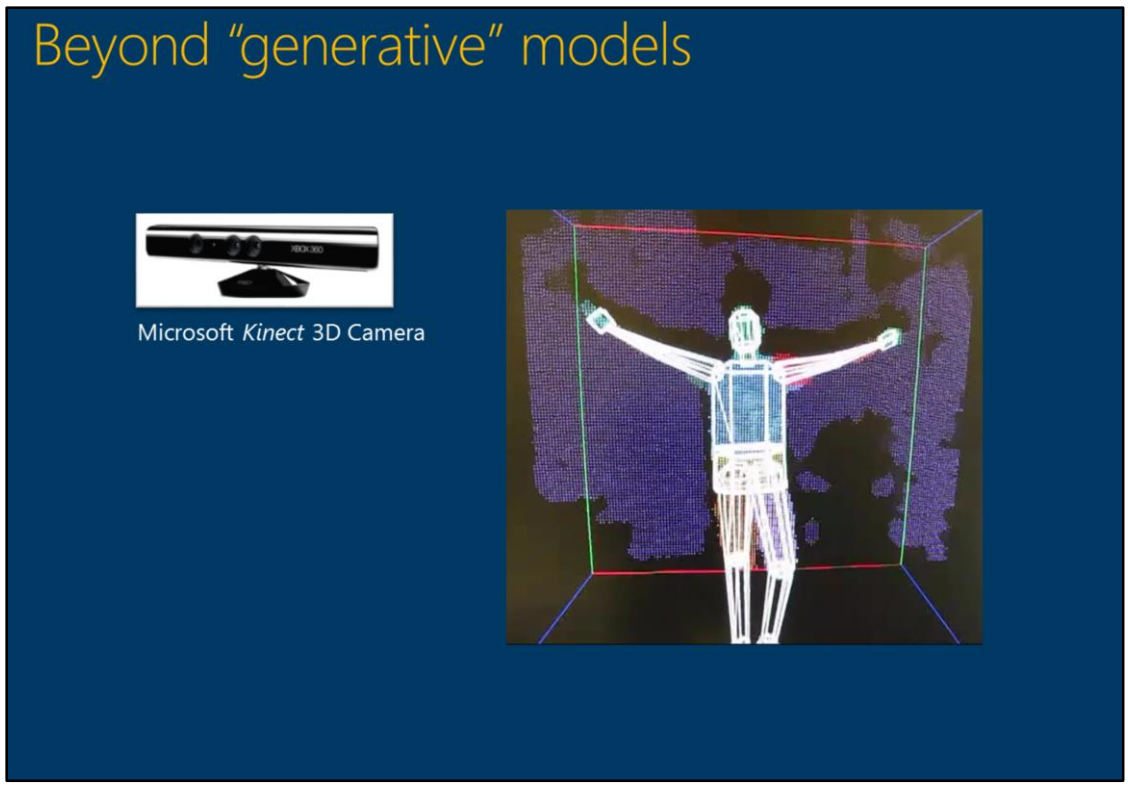
A bird's eye view of the scanned structure of the bookshop. Most of the 3 story bookshop is scanned in a walk-through lasting about 6 minutes. The result is finally represented as a 33 million triangle mesh.



At this point the lecture returns to the modelling and segmentation of shape. The final theme of the lecture is that, powerful though it undoubtedly is, generative modelling – exploring shape hypotheses to find the hypothesis that explains the image data as fully as possible, is not enough on its own for robust tracking of moving shapes.

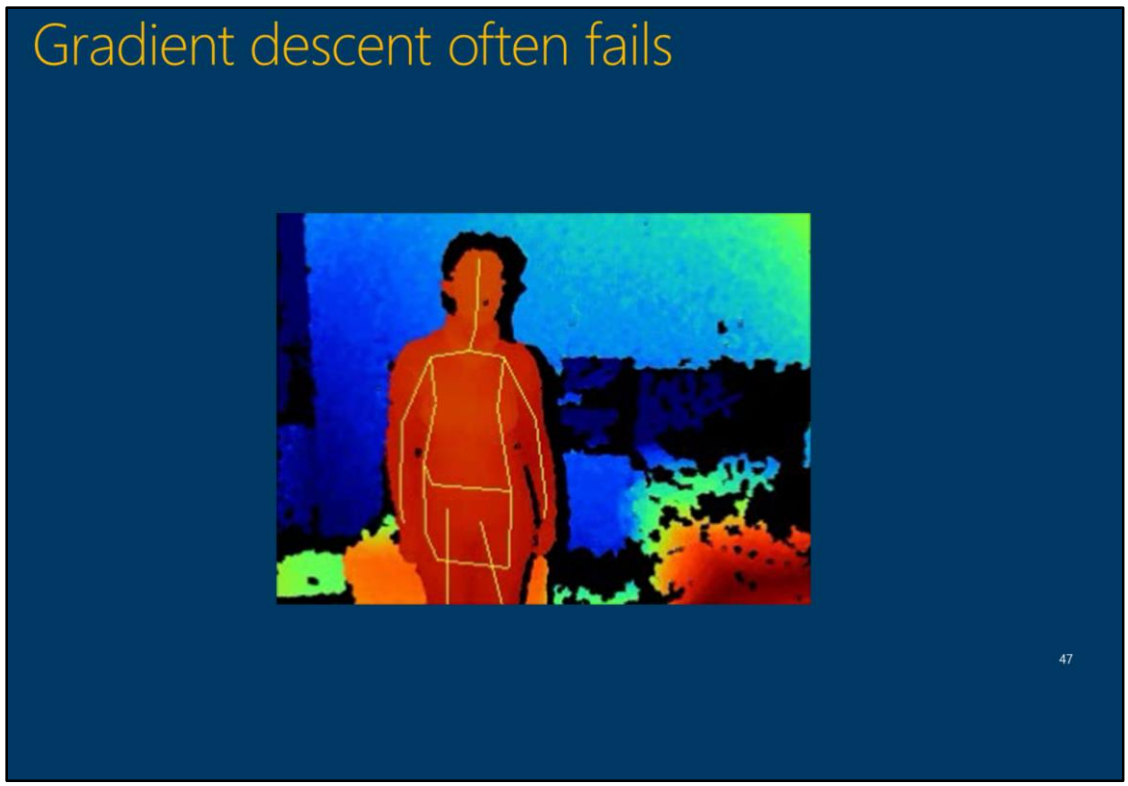
Now in a more complex setting, the aim is to track the detailed motion of a moving human body in three-dimensions. Again the Kinect depth-sensing camera is used to obtain 3D shape data. The model is also more complex than the 2D outline models used earlier, consisting of a jointed, articulated doll-like simulation, like an artists dummy, but simulated by computer graphics. (This is no longer using the level-set formulation.) A generative model to explain the depth-image consists of adjusting the configuration of the dummy to simulate as closely as possible the sensed depth-map.

As a person enters the scene, carrying their depth map into the scene and up to the dummy …



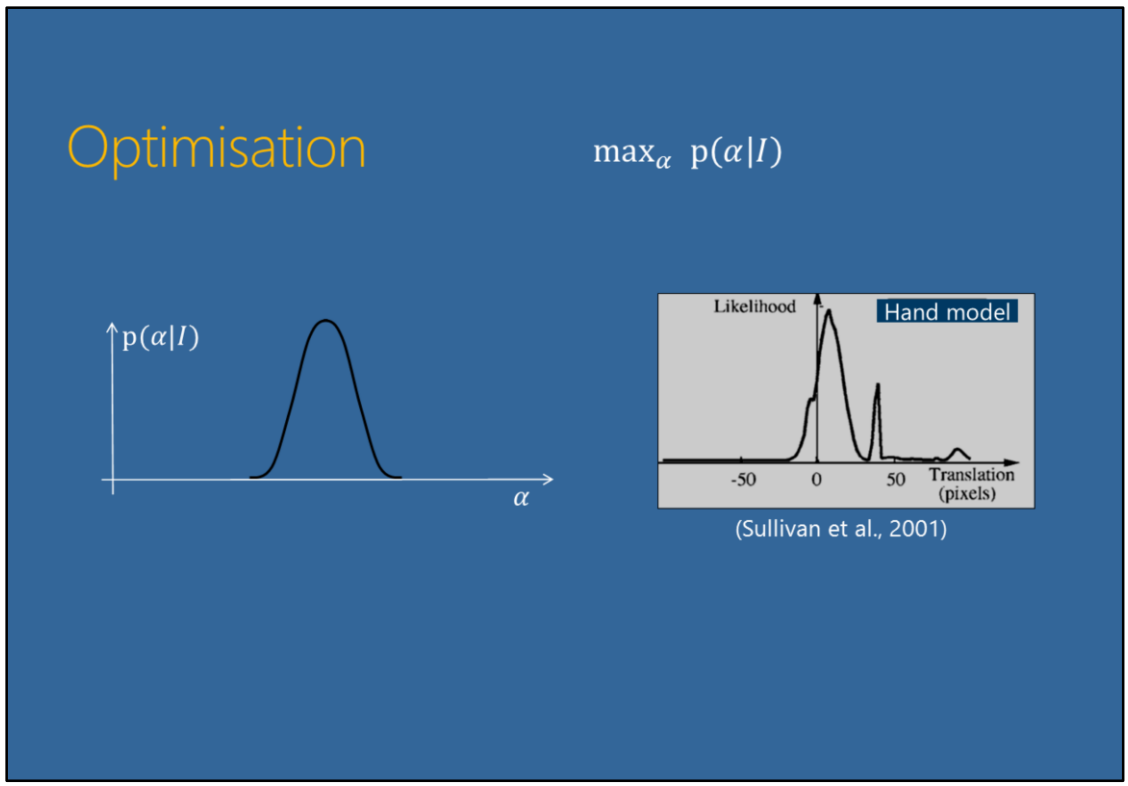
… the depth-map enters the basin of attraction of the model and the dummy locks onto the moving shape. Locking on is achieved by gradient descent on an error function that measures the degree of mismatch between the sensed depth map and the depths simulated from the dummy.

Note that successful locking-on requires a high degree of cooperation and observation from the moving human subject, willingly to walk into the basing of attraction of the model. The cooperative human has to walk, arms outstretched, towards the dummy, watching the screen as carefully as an aeroplane pilot lining up to land on the runway.

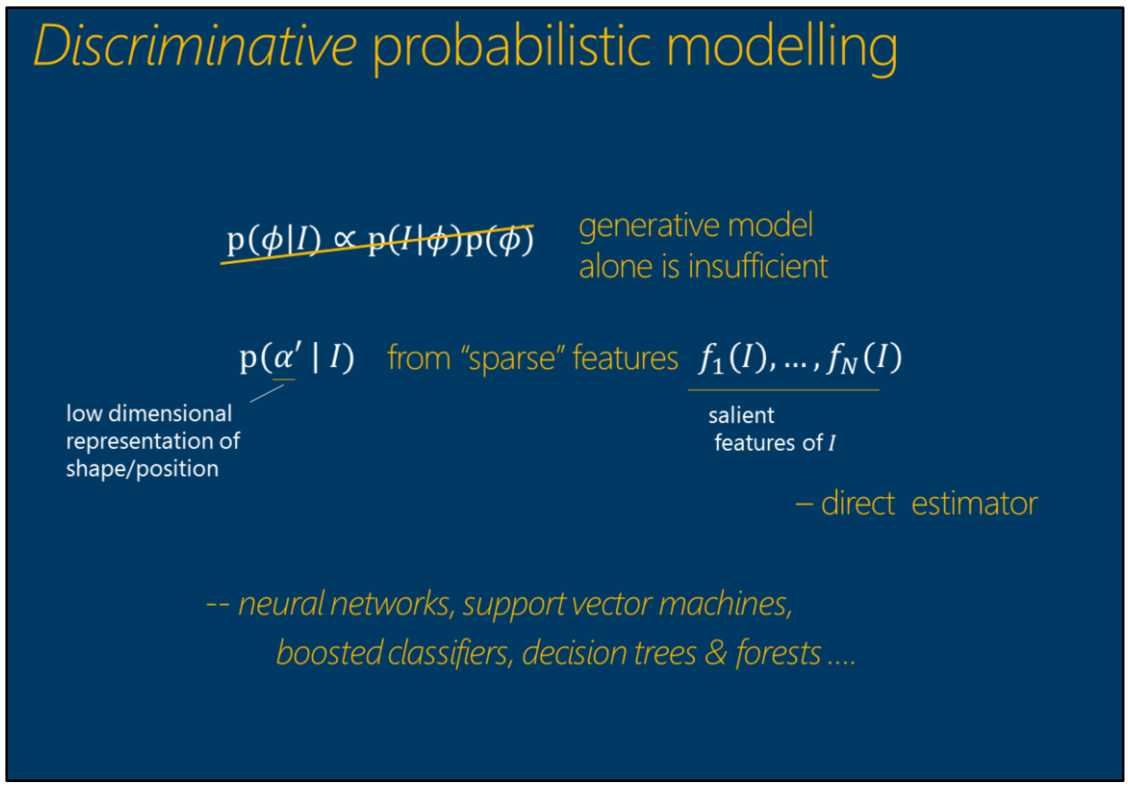


Furthermore, beyond the issue of needing to line up carefully to initialise the model, failures can also occur "in flight". Here the skeleton of the dummy is displayed as yellow curves. At one moment the skeleton is tracking motion successfully ….

… but at the next, an agile motion has thrown the dummy out of its basin of attraction, never to recover.

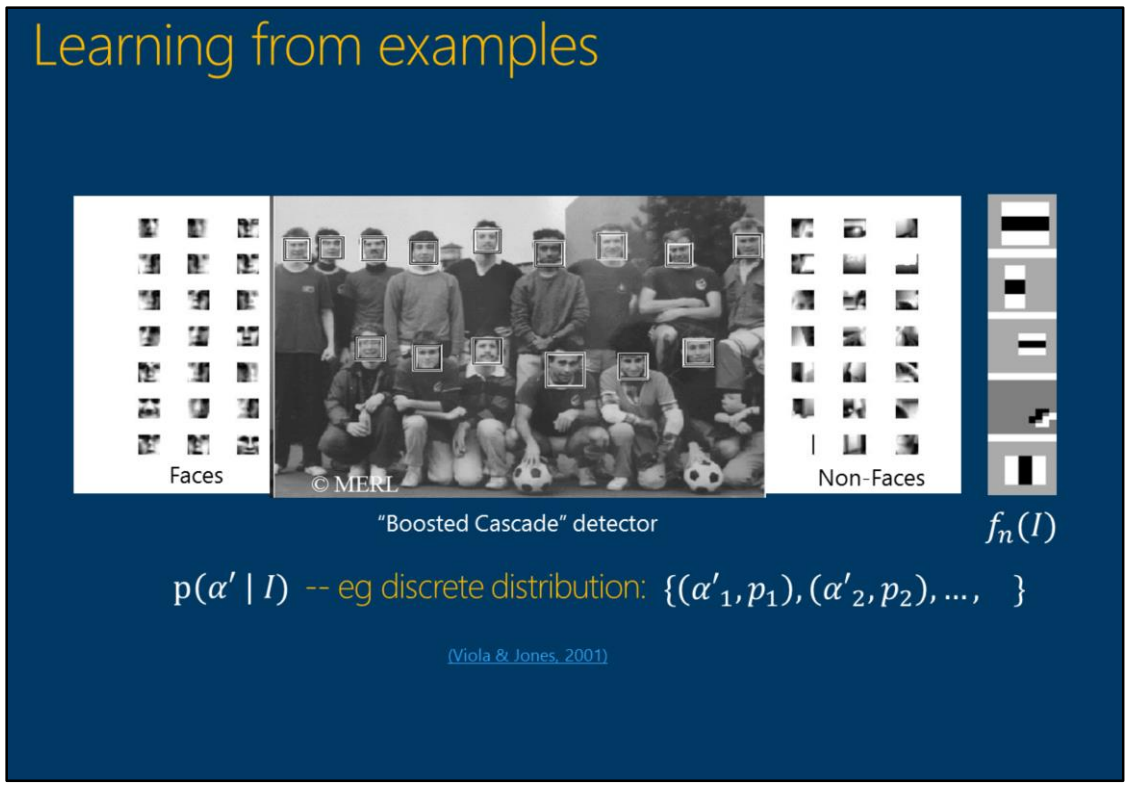


One way of seeing the difficulty with purely generative modelling is to consider this picture of the posterior distribution over $\boldsymbol{\alpha}$, simplified to one dimension. If the posterior were unimodal, as illustrated on the left, then maximising the posterior (minimising the energy) by gradient following would work correctly. But the more usual situation, as shown in measurements from an image of a hand (right), is that the posterior is multi-modal. It is the noise and ambiguity inherent in images that causes the multi-modality.



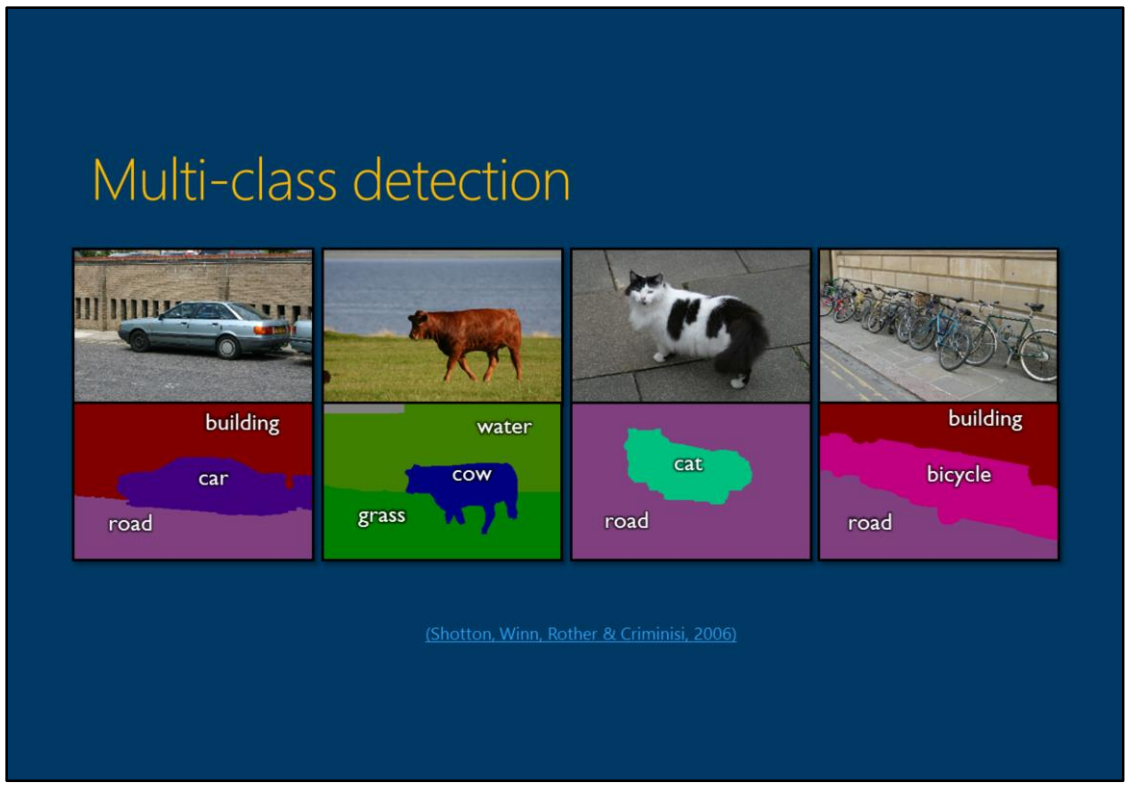
An adjunct to pure generative modelling, whose introduction is essential to achieve robust tracking, is discriminative modelling. In the discriminative paradigm we *start over* with the building of a model for recognition. It really shares little or no machinery with the generative model, but ultimately has to be combined (currently in an ad hoc fashion) with generative modelling, to achieve a robust and precise tracking system overall. The discriminative element contributes robustness, and the generative element ensures precision and detail.

The discriminative element uses a typically even lower dimensional representation α' of shape and/or position, which however is not used as a hypothesis in a Bayesian model. Instead, values of α', or a simple posterior distribution for α' is estimated directly from the image I. Even then, unlike the generative case, discriminative modelling does not use the whole of I but just a few particularly salient features $f_1(I), \ldots, f_N(I)$. These features are designed to be computed efficiently, and the choice of the most salient components is made as part of an extended learning procedure.



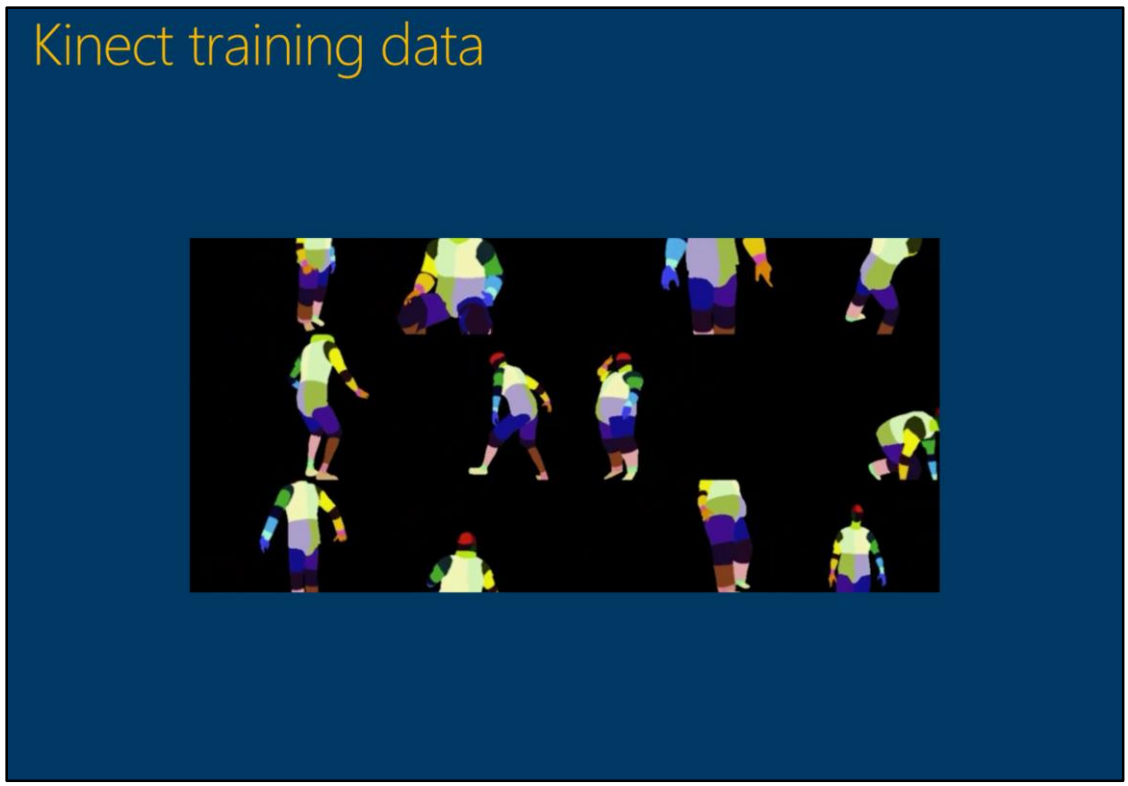
A seminal example of discriminative model is the face detector shown above, developed at Mitsubishi laboratories, which was so revolutionary and effective that it was rapidly incorporated in consumer cameras, to help automatically compose good shots of people. The posterior distribution for α' is computed directly from the salient features, which themselves are computed by placing masks of weights (illustrated on the right), at various random locations in the image, and computing the weighted sum of intensity values under each mask. Initially this is done with large numbers of randomly generated masks, and the few of those that are most salient are selected by the learning process – known as *boosting*.

(Viola & Jones, 2001)



Quite a different kind of discriminative model is shown here, and this is the kind used to assist the generative model in 3D human body tracking with the Kinect camera. The problem here is to label each pixel of an image with one of a number (20 in this example) of classes, according to object type. A classifier the outputs a probability distribution over the 20 classes is applied to each pixel. The probabilities are computed directly from image features, which can be ones similar to those used in the face-detection example.

(Shotton, Winn, Rother & Criminisi, 2006)



To apply this methodology to human body motion, the body is treated as an assembly of parts, and each as one of a number (31 in this example) of possible parts. To build the classifier, it is necessary to assemble a training set of labelled bodies, cover all likely body shapes and configurations.

To do this, a number of examples ….

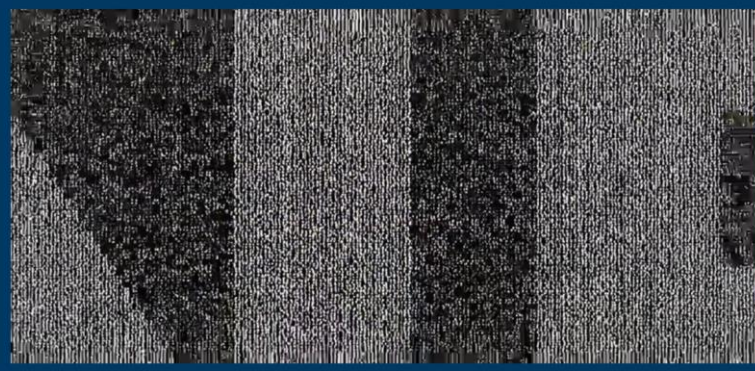
Kinect training data

… quite a number ….

Kinect training data

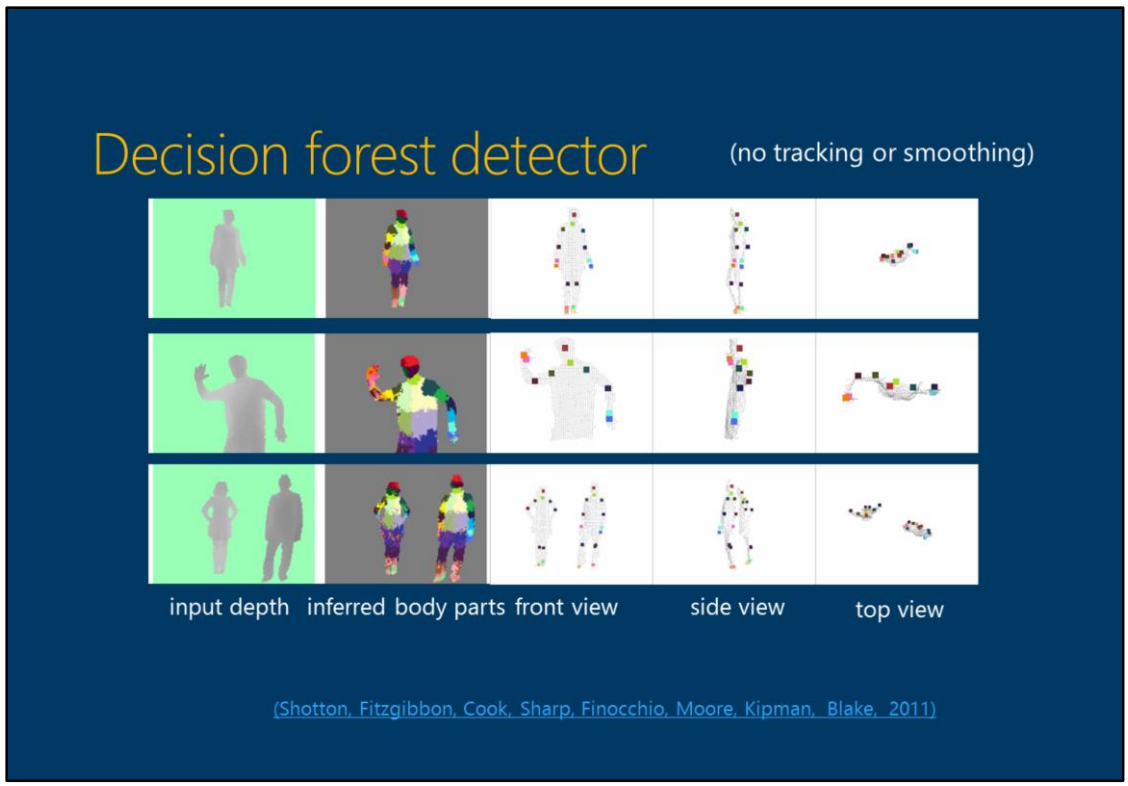
… a substantial number …

Kinect training data



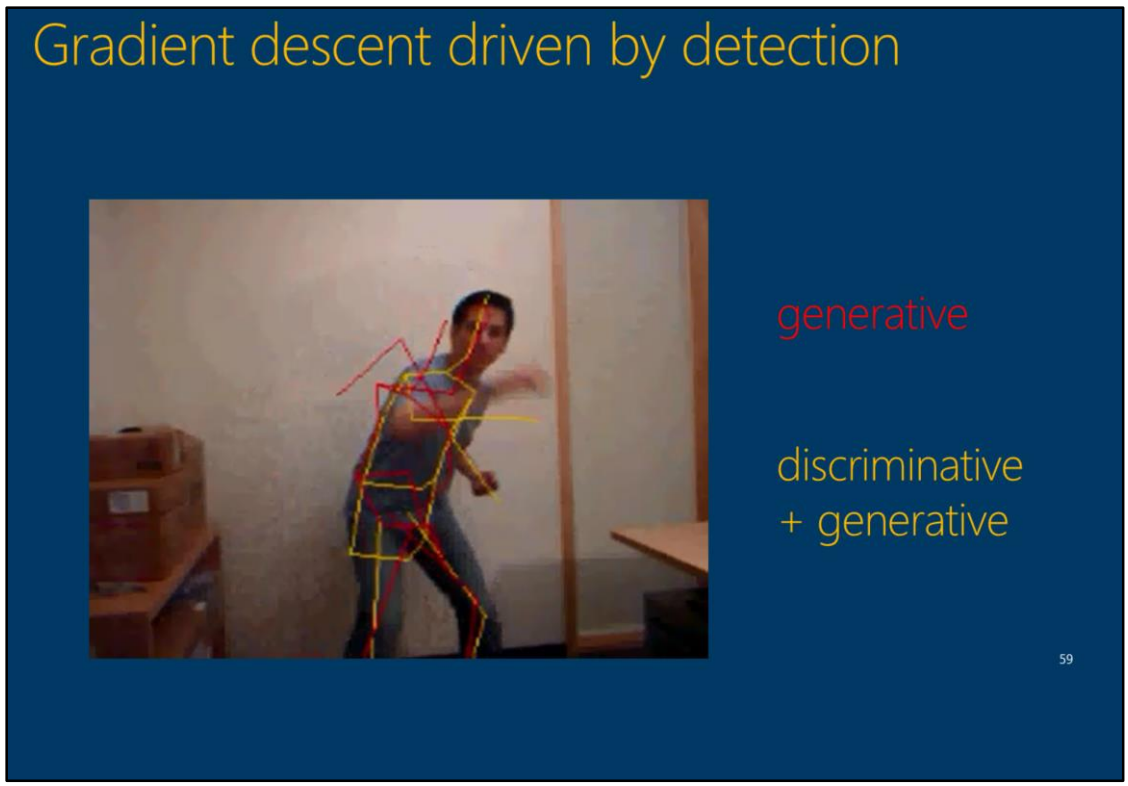
… really surprisingly many ….

… in fact, over a million labelled examples were needed to achieve good classification performance.



And the result is a system that yields labelled depth maps, which can then be simplified to give estimates of the positions of the major joints of the body, and those positions, in turn, initialise the generative model.

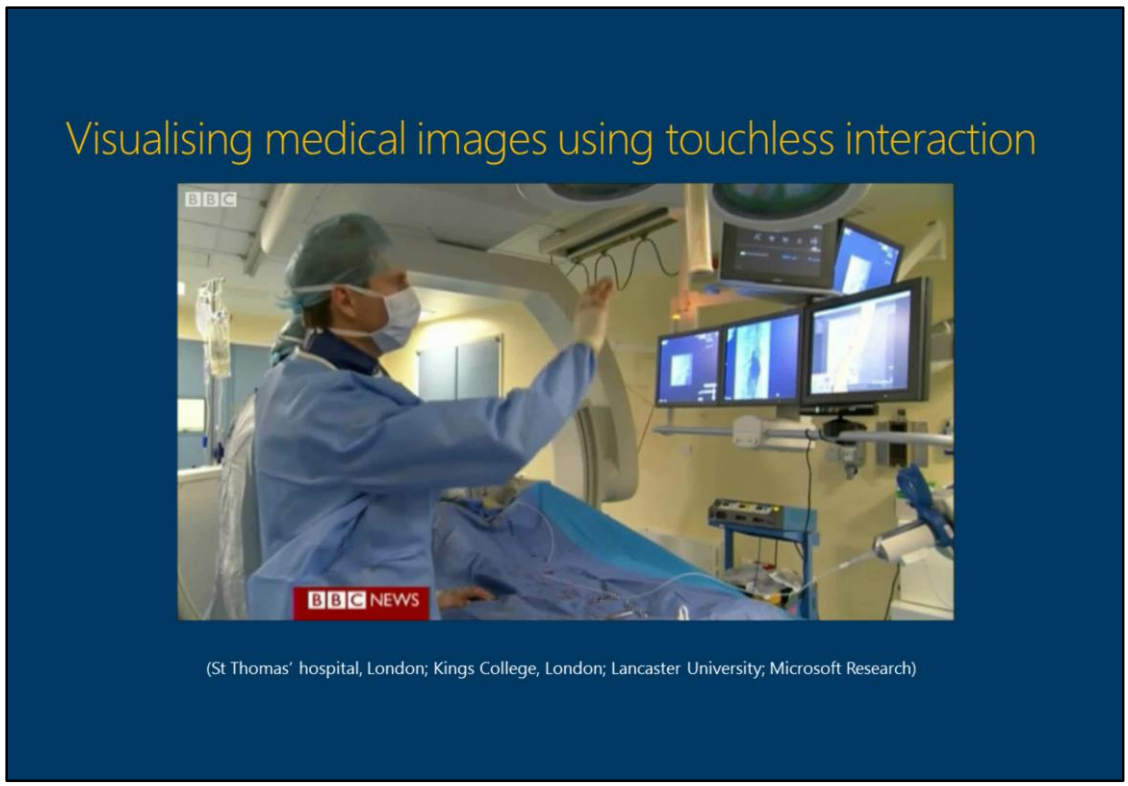
(Shotton, Fitzgibbon, Cook, Sharp, Finocchio, Moore, Kipman, Blake, 2011)



Here is an illustration of the discriminative and generative models, in combination, succeeding where the generative model, on its own, does not.

Hands-free interaction

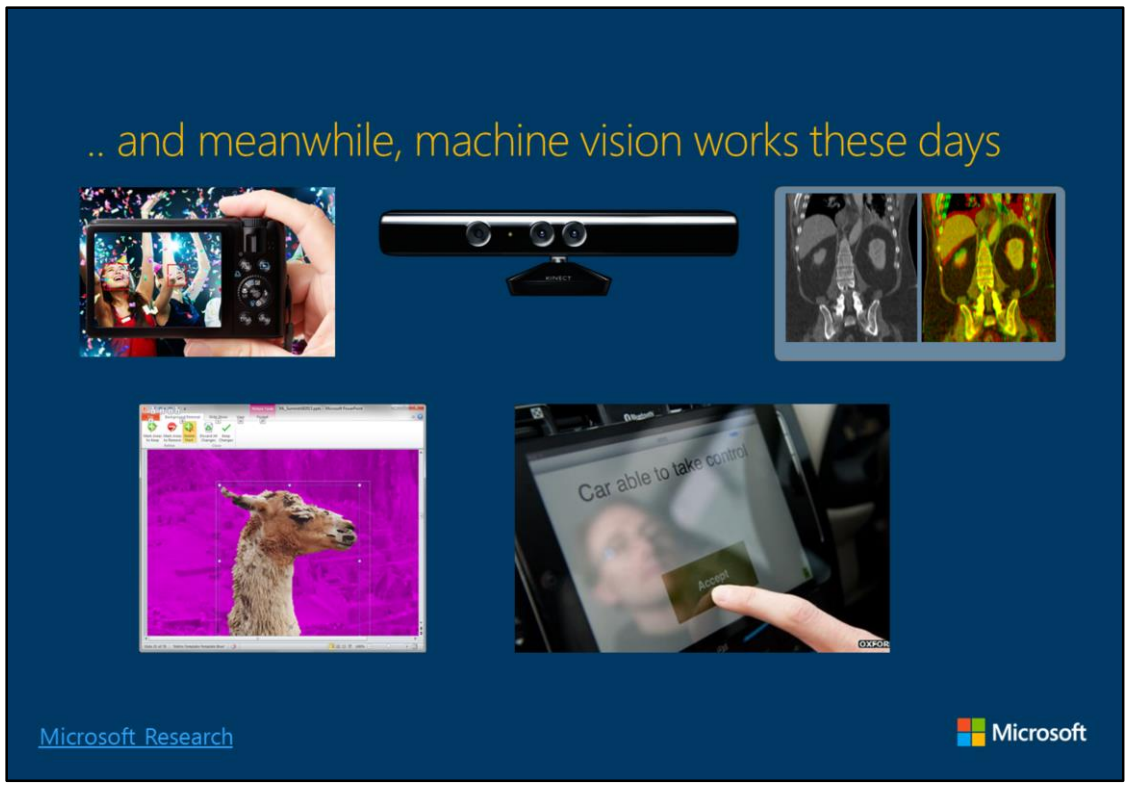
Kinect's 3D body motion tracking system is now widely used for gaming.



Kinect 3D body motion tracking is also finding many other applications, for example in robotics, and in medicine as illustrated here.

The best future vision systems are likely to combine the generative and the discriminative.

Currently, the discriminative and generative sub-systems are typically "bolted" together, but I am optimistic that a truly elegant and powerful synthesis of the generative with the discriminative will be found in future – one that gives a consistent probabilistic account of the combined system, and does so with the necessary computational efficiency.



In the meantime it is a cause for some celebration that there are machines working on a commercial scale that incorporate machine vision. This is in large measure thanks to mathematical – geometric and probabilistic – principles, mathematical thinking that is making the world safer, healthier and more interesting, by means of machines that see.

Microsoft Research