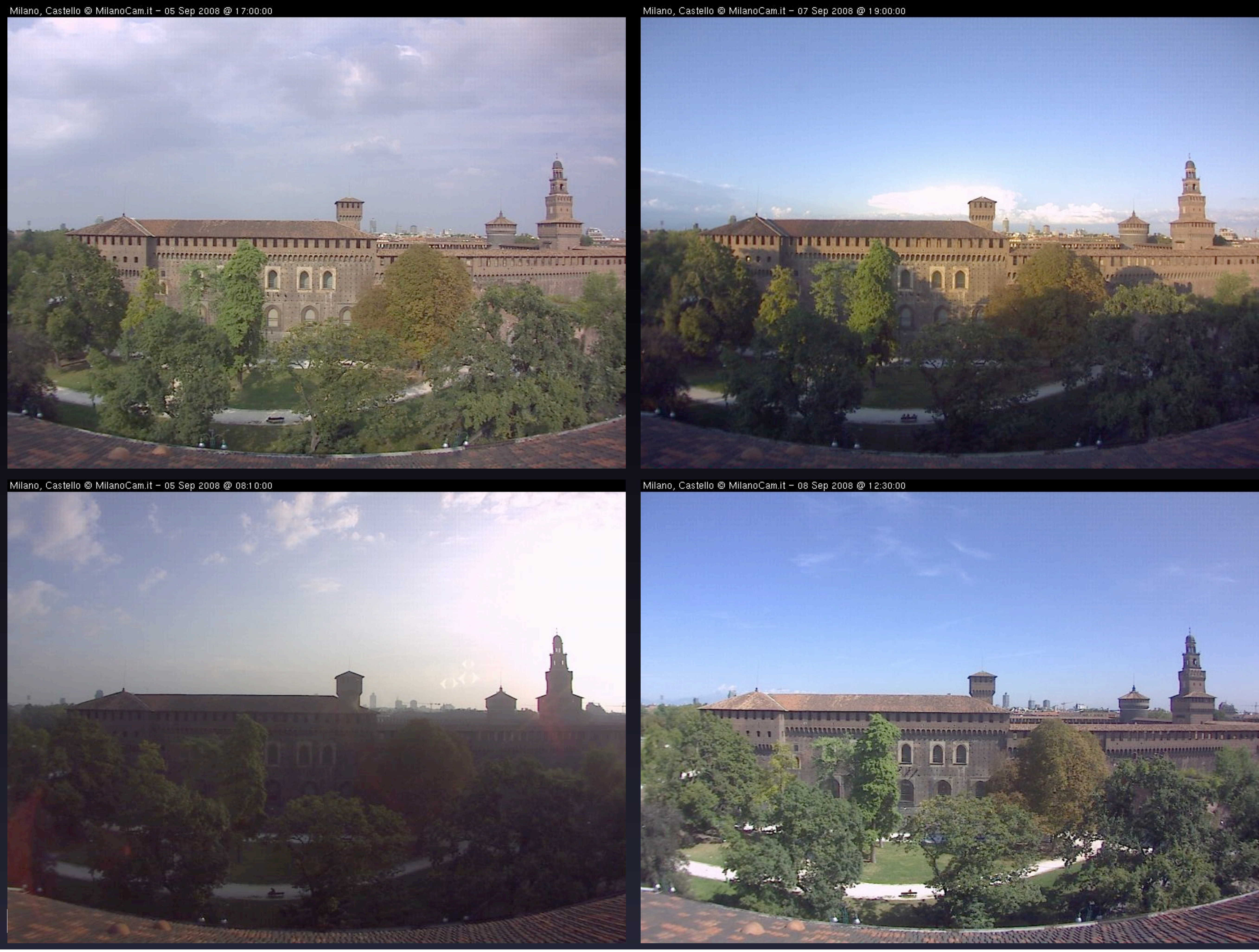
Estimating Natural Illumination from a Single Outdoor Image

Jean-François Lalonde, Alexei A. Efros, and Srinivasa G. Narasimhan
School of Computer Science, Carnegie Mellon University

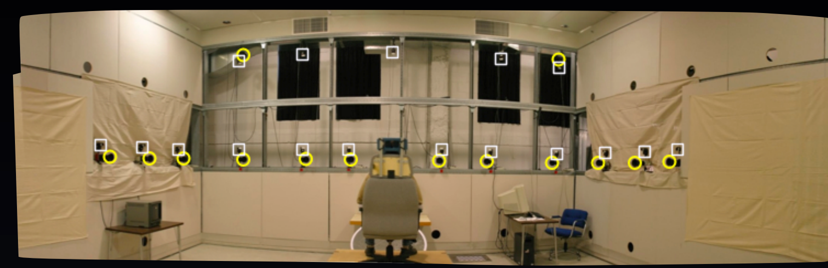
[Lalonde, Efros, and Narasimhan, ICCV 2009]



It's no secret that the appearance of scenes strongly depend on the illumination conditions. For example, in these images, everything is kept constant except the illumination, and yet the pixel values are completely different. Because appearance can vary so much, estimating illumination from images is a very important task in computer vision.

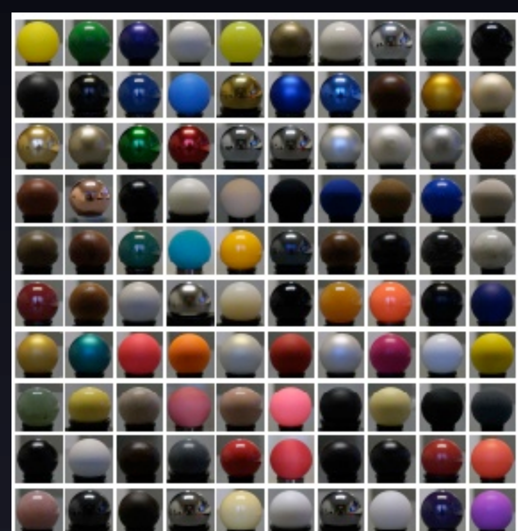
Lighting in the lab

[Georghiades *et al.*, PAMI '01]

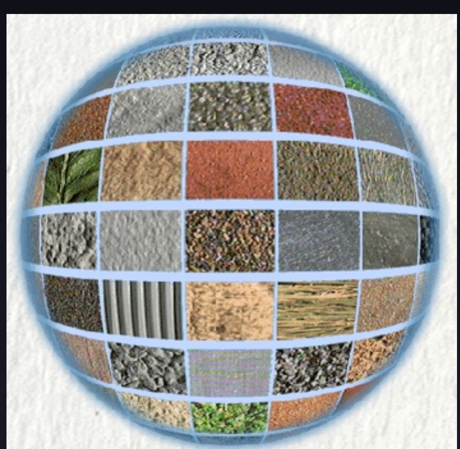


[Gross *et al.*, F&G '08]

[Debevec *et al.*, SIGGRAPH '02]

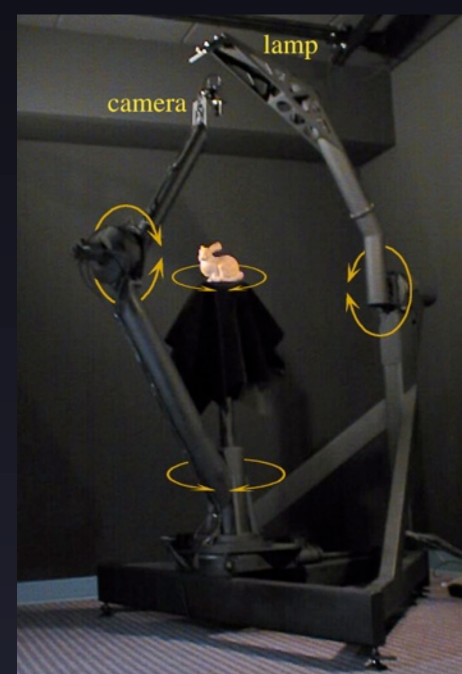


[Matusik *et al.*, SIGGRAPH '03]



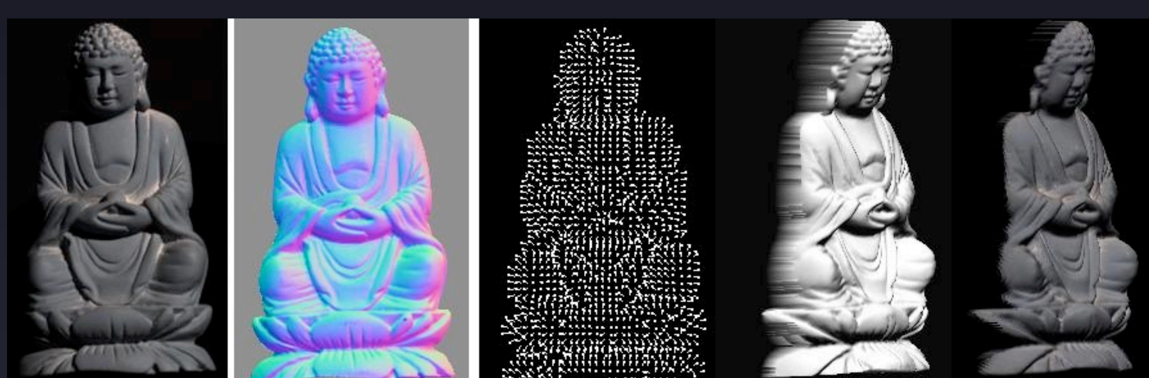
[Dana *et al.*, TOG '99]

[Ramamoorthi & Hanrahan, SIGGRAPH '01]



[Wood *et al.*, SIGGRAPH '00]

[Belhumeur *et al.*, ECCV '96]



[Hertzmann & Seitz, PAMI '05]

[Lalonde, Efros, and Narasimhan, ICCV 2009]

Indeed, a lot of work has been done in this area, and here I'm just showing a small sampling of it. One thing to note here: all of this is done in the lab, with tightly controlled conditions. But what about images in the wild, outside of the lab?

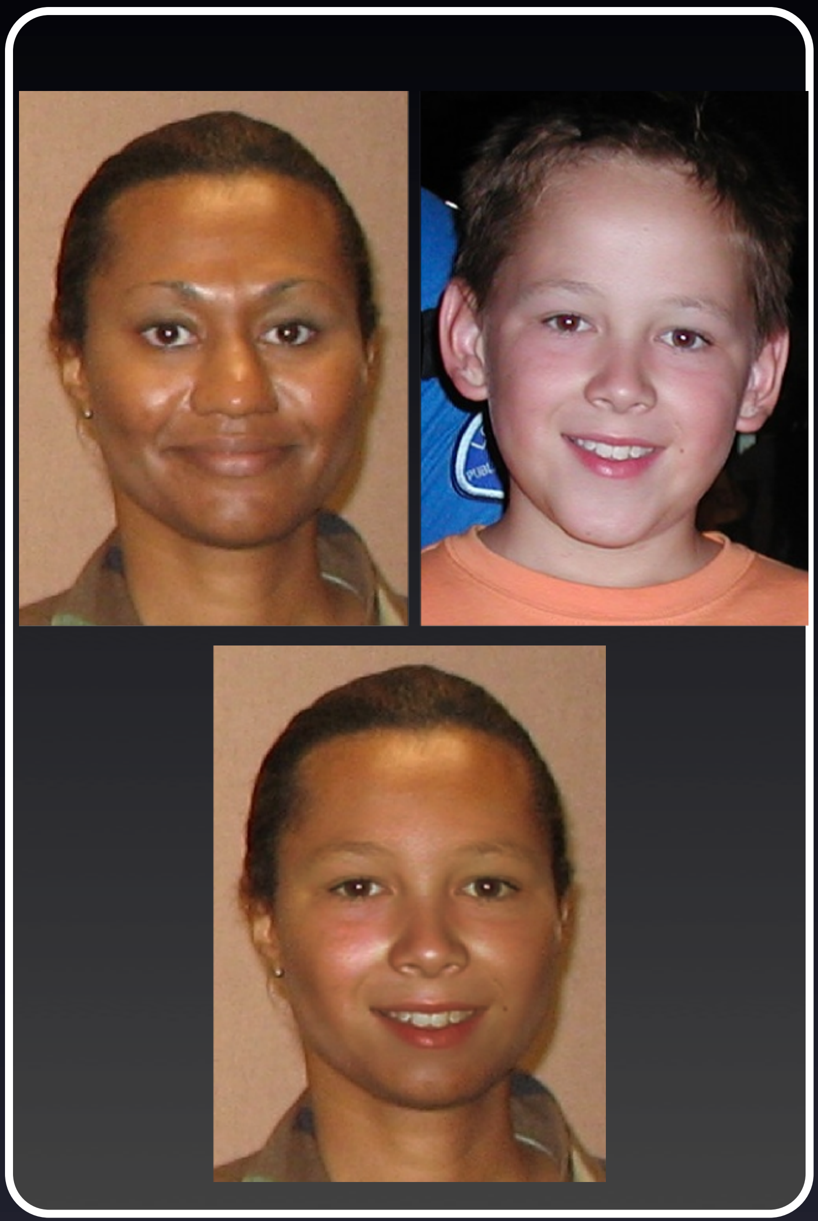
Lighting in the wild

[Lalonde, Efros, and Narasimhan, ICCV 2009]

Well, there's actually very little work dealing with illumination in real settings. Recently, there's been a few papers dealing with real consumer imagery but they were focusing either on specific objects such as faces, or dealt with many images like in webcam sequences or image collections.
But what about single images?

Lighting in the wild

Specific objects



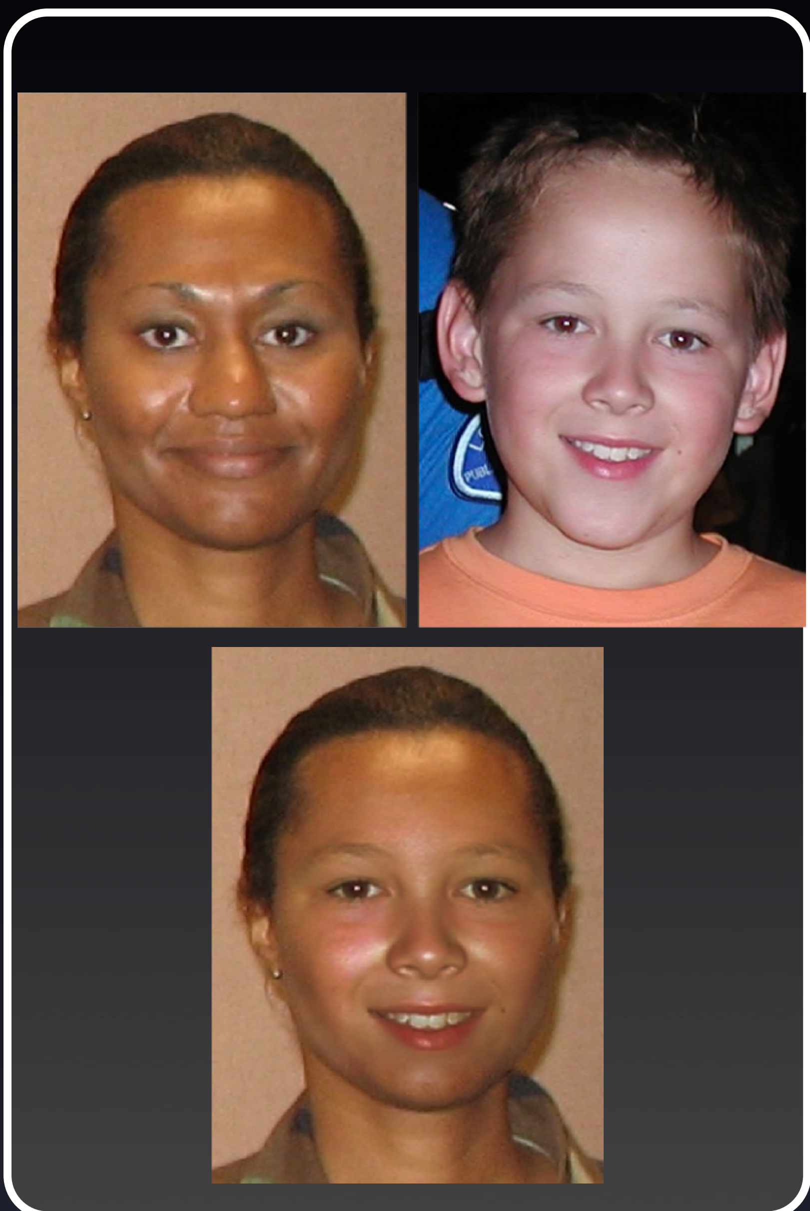
[Bitouk *et al.*, SIGGRAPH '08]

[Lalonde, Efros, and Narasimhan, ICCV 2009]

Well, there's actually very little work dealing with illumination in real settings. Recently, there's been a few papers dealing with real consumer imagery but they were focusing either on specific objects such as faces, or dealt with many images like in webcam sequences or image collections.
But what about single images?

Lighting in the wild

Specific objects



[Bitouk *et al.*, SIGGRAPH '08]

Image sequences

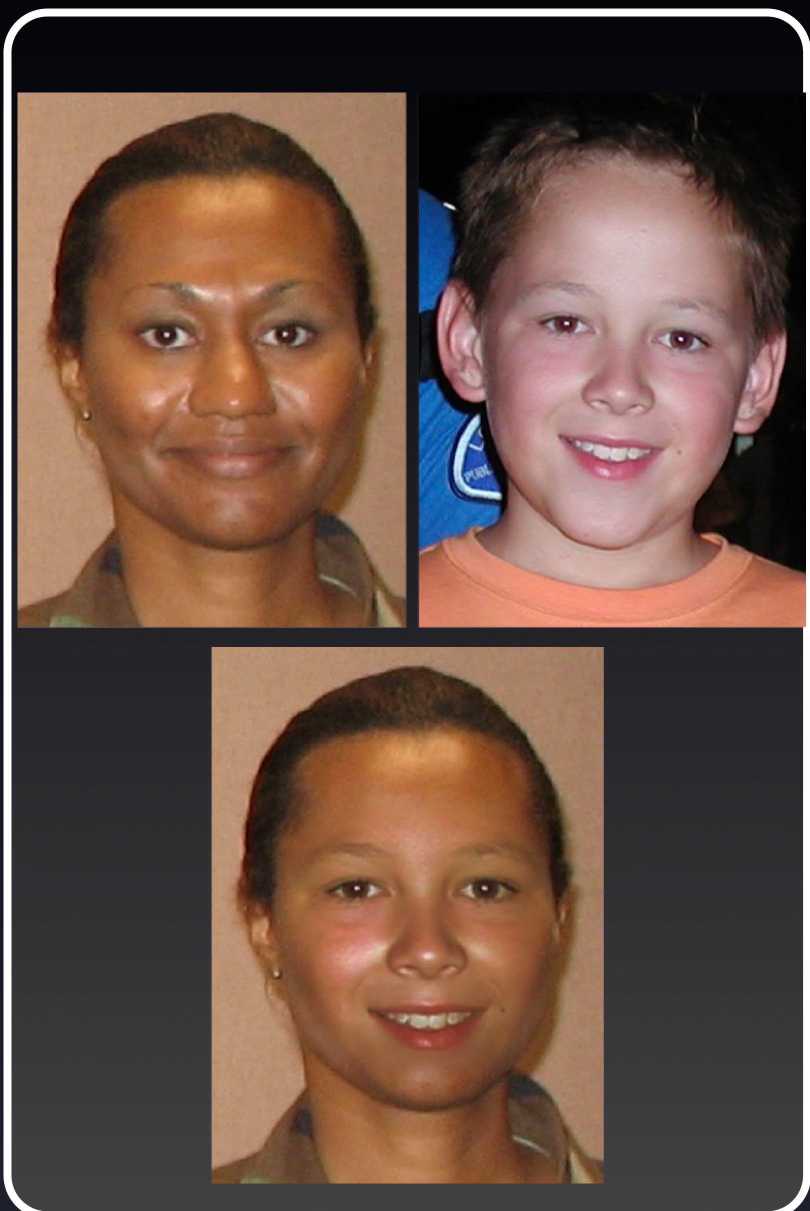
[Sunkavalli *et al.*, CVPR '08]

[Lalonde, Efros, and Narasimhan, ICCV 2009]

Well, there's actually very little work dealing with illumination in real settings. Recently, there's been a few papers dealing with real consumer imagery but they were focusing either on specific objects such as faces, or dealt with many images like in webcam sequences or image collections.
But what about single images?

Lighting in the wild

Specific objects



[Bitouk *et al.*, SIGGRAPH '08]

Image sequences

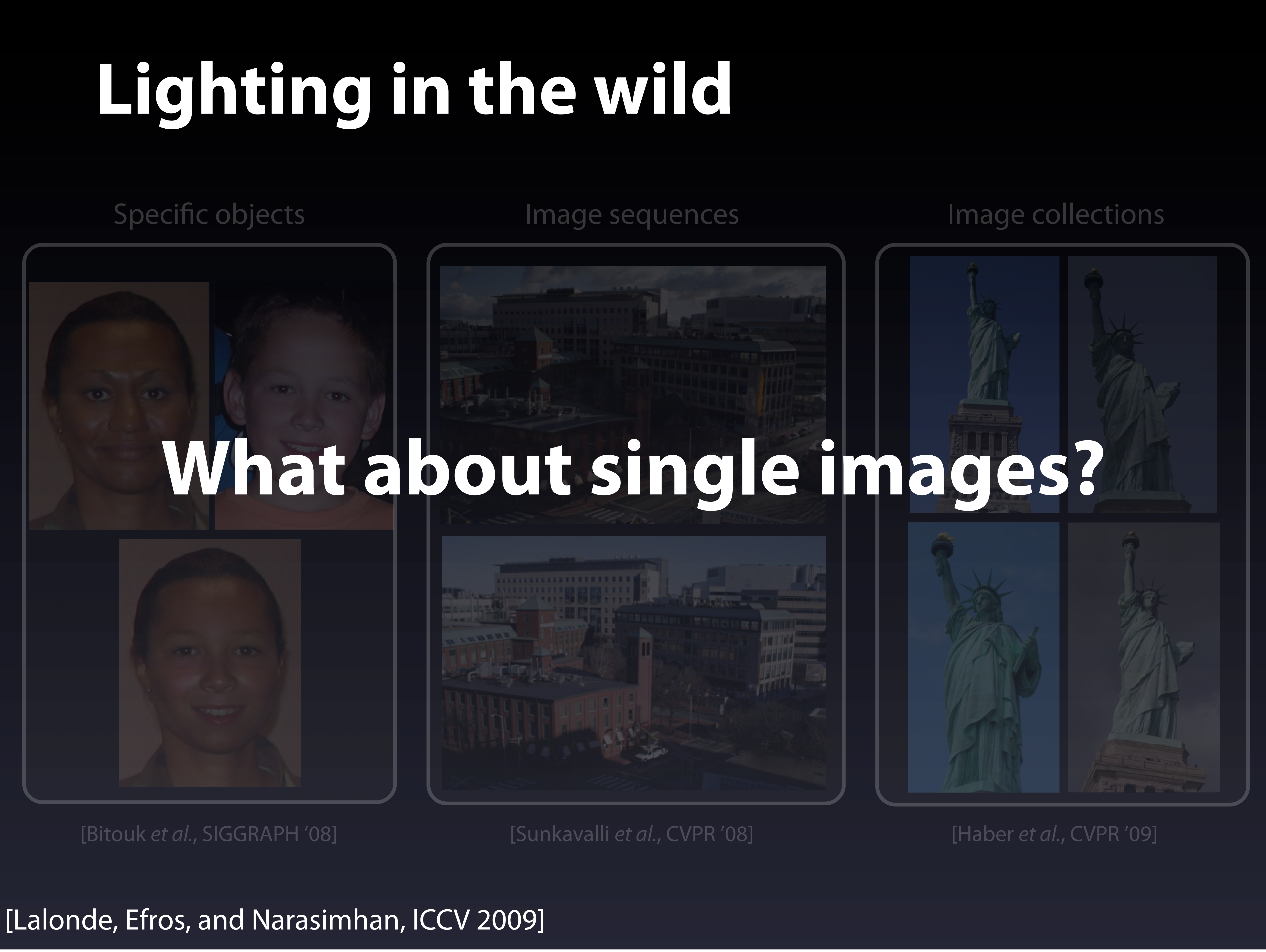
[Sunkavalli *et al.*, CVPR '08]

Image collections

[Haber *et al.*, CVPR '09]

[Lalonde, Efros, and Narasimhan, ICCV 2009]

Well, there's actually very little work dealing with illumination in real settings. Recently, there's been a few papers dealing with real consumer imagery but they were focusing either on specific objects such as faces, or dealt with many images like in webcam sequences or image collections.
But what about single images?



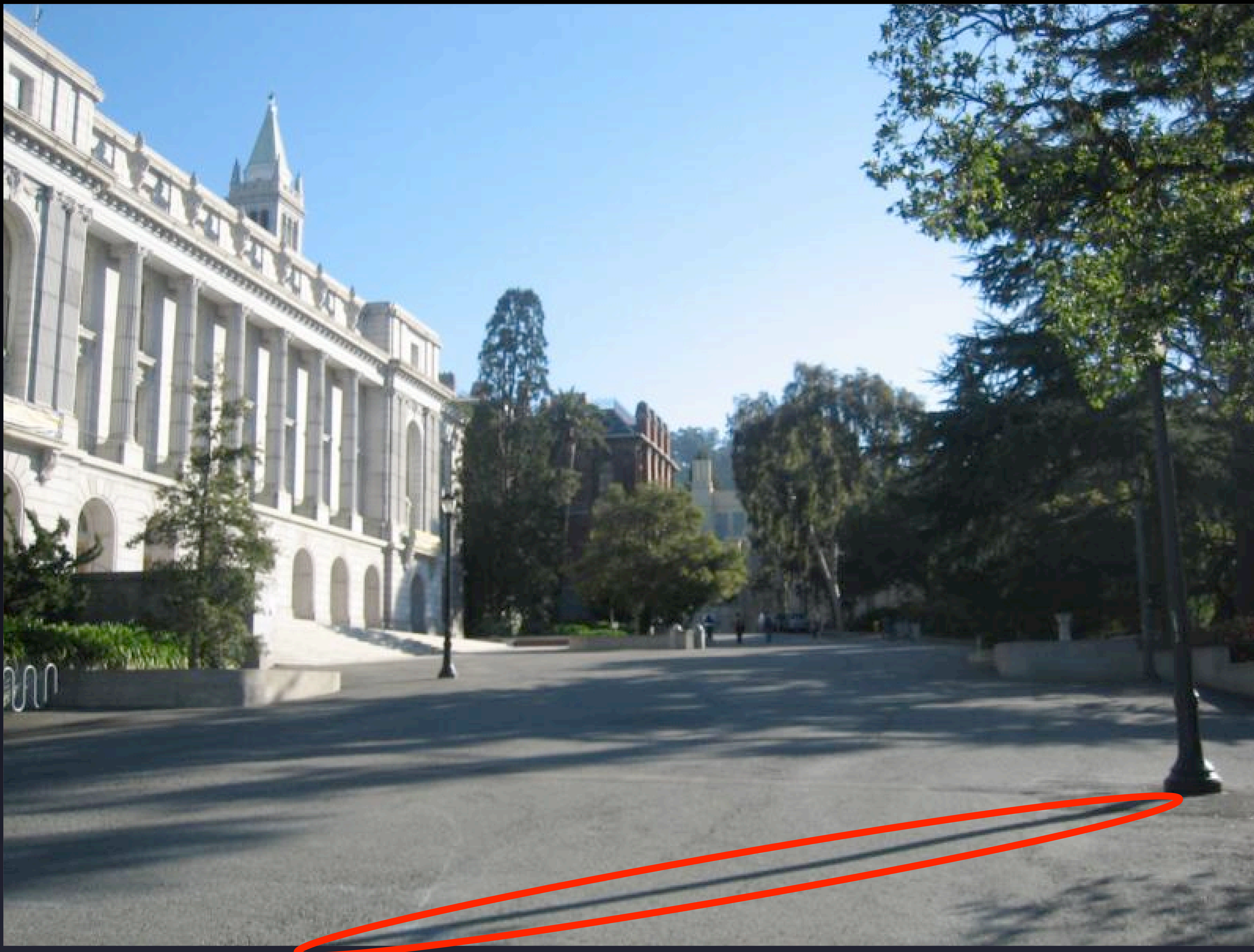
Well, there's actually very little work dealing with illumination in real settings. Recently, there's been a few papers dealing with real consumer imagery but they were focusing either on specific objects such as faces, or dealt with many images like in webcam sequences or image collections.
But what about single images?

[Lalonde, Efros, and Narasimhan, ICCV 2009]

Well this is a hugely under-constrained problem: we don't know the capture conditions, material properties, scene geometry, nor illumination conditions. Yet when we look at this image, we're pretty sure that the sun is coming from the right, because we're able to exploit the different effects that are caused by the illumination conditions: shadows, sky, shading. Unfortunately, these effects have been largely ignored in computer vision.

In this talk, I will show that we can extract these 3 cues from images in order to automatically estimate the illumination conditions from images. In particular we'll focus on the relative position of the sun with respect to the camera, although in the paper we do estimate whether the sun is being occluded (by a cloud) or not.

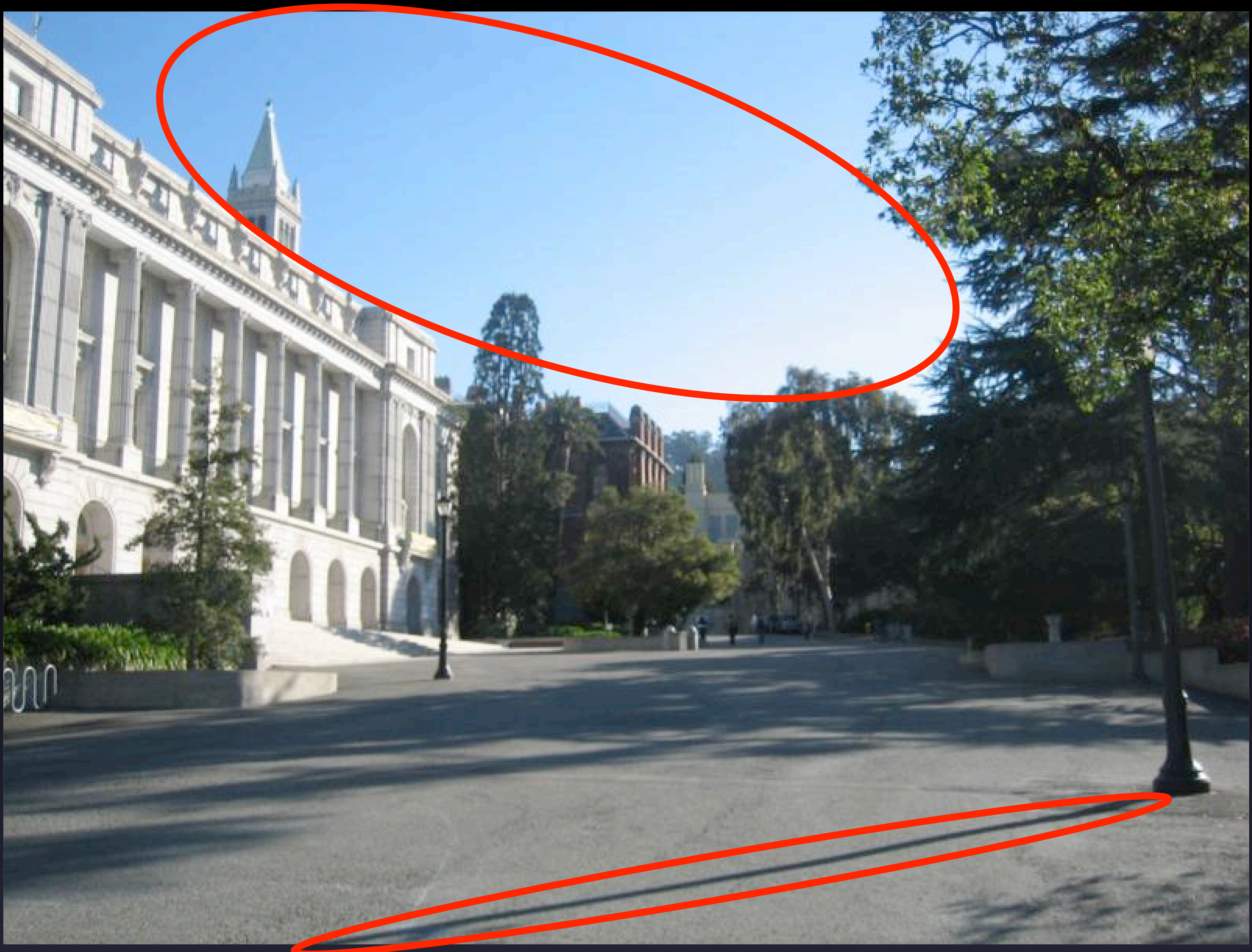
Of course, these cues can be very weak, they might not always be available, but together, they still provide us with some sense of lighting direction, and it is this qualitative sense of direction that we're after in the paper.



[Lalonde, Efros, and Narasimhan, ICCV 2009]

Well this is a hugely under-constrained problem: we don't know the capture conditions, material properties, scene geometry, nor illumination conditions. Yet when we look at this image, we're pretty sure that the sun is coming from the right, because we're able to exploit the different effects that are caused by the illumination conditions: shadows, sky, shading. Unfortunately, these effects have been largely ignored in computer vision.
In this talk, I will show that we can extract these 3 cues from images in order to automatically estimate the illumination conditions from images. In particular we'll focus on the relative position of the sun with respect to the camera, although in the paper we do estimate whether the sun is being occluded (by a cloud) or not.

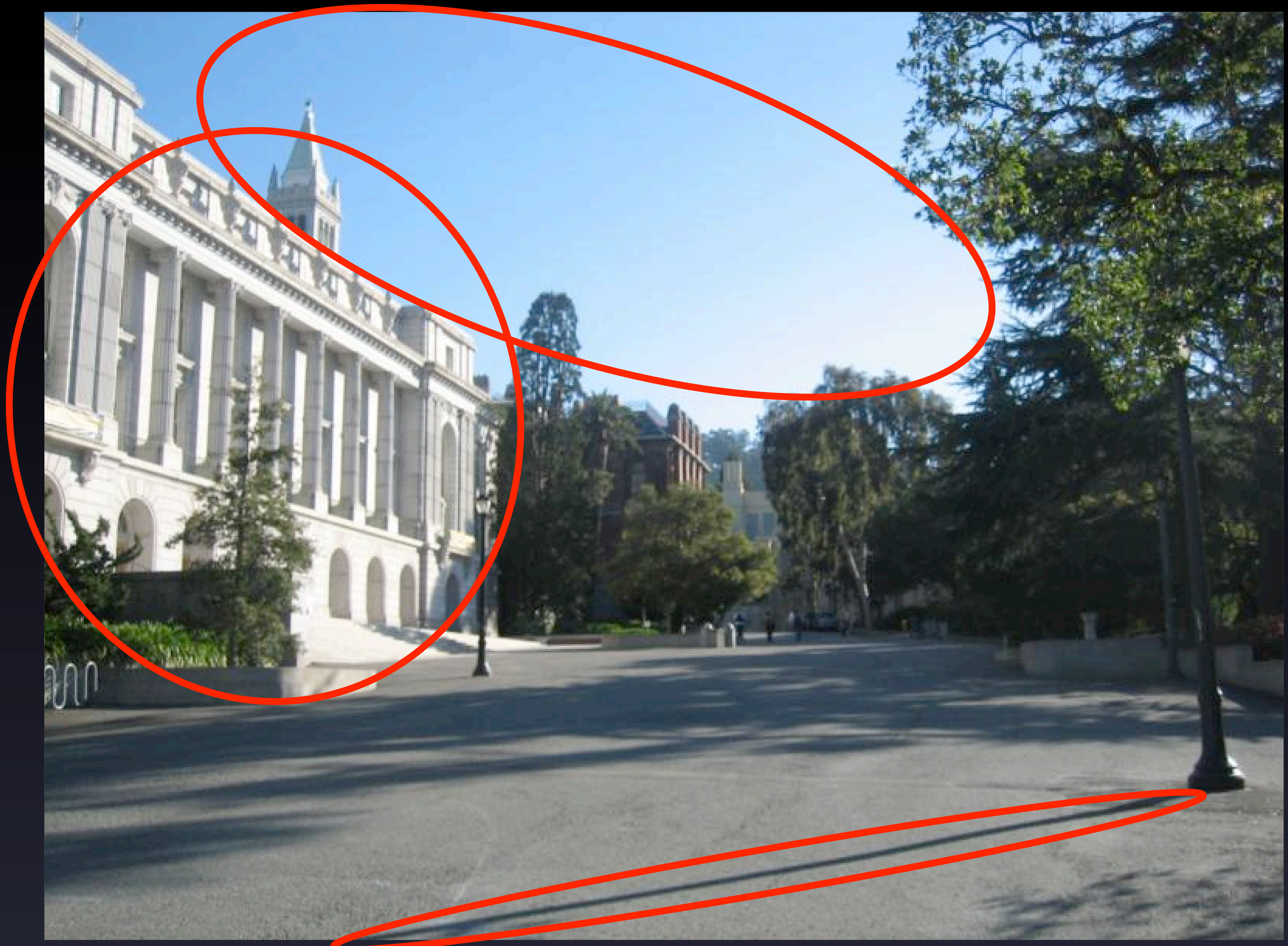
Of course, these cues can be very weak, they might not always be available, but together, they still provide us with some sense of lighting direction, and it is this qualitative sense of direction that we're after in the paper.



[Lalonde, Efros, and Narasimhan, ICCV 2009]

Well this is a hugely under-constrained problem: we don't know the capture conditions, material properties, scene geometry, nor illumination conditions. Yet when we look at this image, we're pretty sure that the sun is coming from the right, because we're able to exploit the different effects that are caused by the illumination conditions: shadows, sky, shading. Unfortunately, these effects have been largely ignored in computer vision.
In this talk, I will show that we can extract these 3 cues from images in order to automatically estimate the illumination conditions from images. In particular we'll focus on the relative position of the sun with respect to the camera, although in the paper we do estimate whether the sun is being occluded (by a cloud) or not.

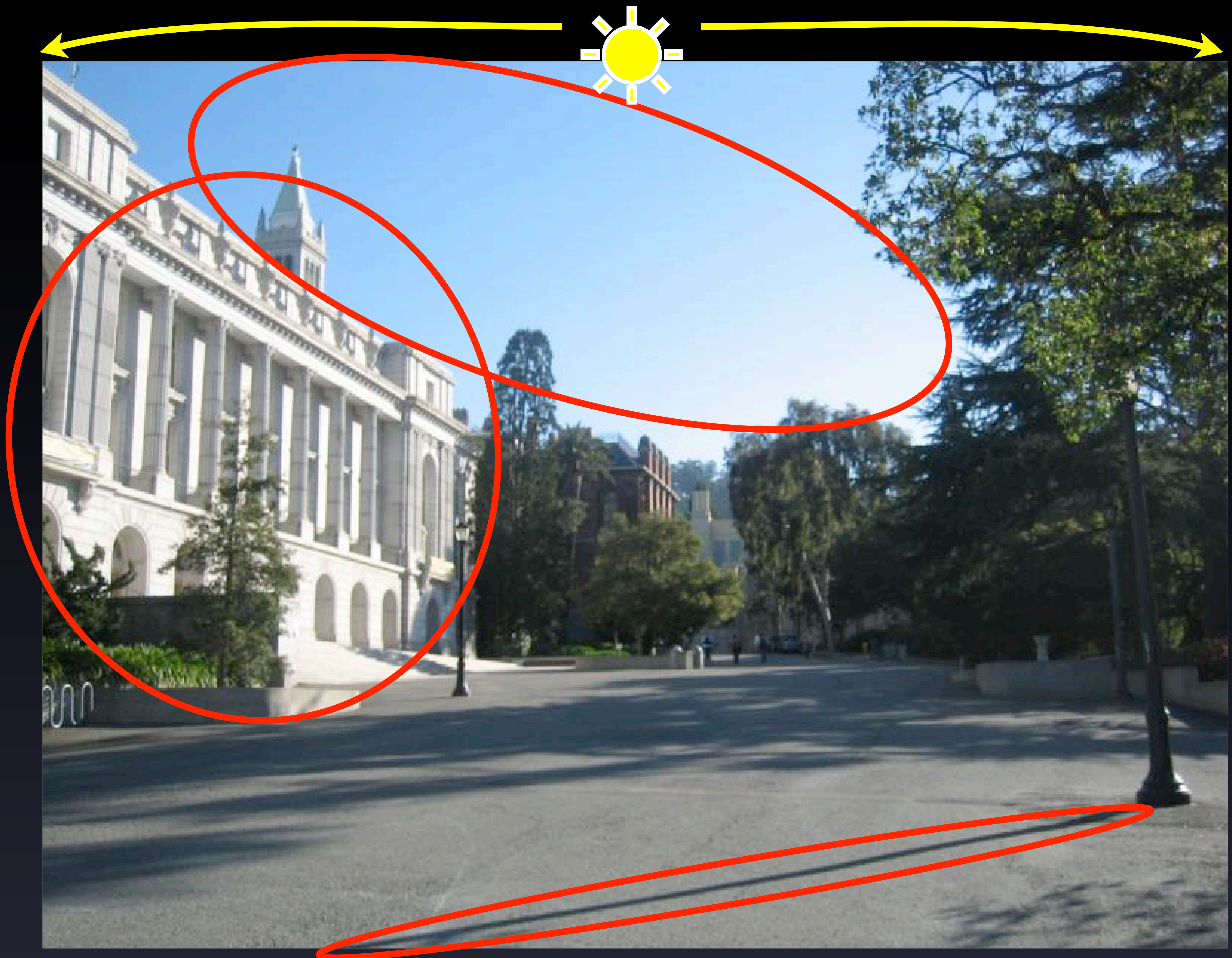
Of course, these cues can be very weak, they might not always be available, but together, they still provide us with some sense of lighting direction, and it is this qualitative sense of direction that we're after in the paper.



[Lalonde, Efros, and Narasimhan, ICCV 2009]

Well this is a hugely under-constrained problem: we don't know the capture conditions, material properties, scene geometry, nor illumination conditions. Yet when we look at this image, we're pretty sure that the sun is coming from the right, because we're able to exploit the different effects that are caused by the illumination conditions: shadows, sky, shading. Unfortunately, these effects have been largely ignored in computer vision.
In this talk, I will show that we can extract these 3 cues from images in order to automatically estimate the illumination conditions from images. In particular we'll focus on the relative position of the sun with respect to the camera, although in the paper we do estimate whether the sun is being occluded (by a cloud) or not.

Of course, these cues can be very weak, they might not always be available, but together, they still provide us with some sense of lighting direction, and it is this qualitative sense of direction that we're after in the paper.



[Lalonde, Efros, and Narasimhan, ICCV 2009]

Well this is a hugely under-constrained problem: we don't know the capture conditions, material properties, scene geometry, nor illumination conditions. Yet when we look at this image, we're pretty sure that the sun is coming from the right, because we're able to exploit the different effects that are caused by the illumination conditions: shadows, sky, shading. Unfortunately, these effects have been largely ignored in computer vision.
In this talk, I will show that we can extract these 3 cues from images in order to automatically estimate the illumination conditions from images. In particular we'll focus on the relative position of the sun with respect to the camera, although in the paper we do estimate whether the sun is being occluded (by a cloud) or not.

Of course, these cues can be very weak, they might not always be available, but together, they still provide us with some sense of lighting direction, and it is this qualitative sense of direction that we're after in the paper.

[Lalonde, Efros, and Narasimhan, ICCV 2009]

But before we talk about how we compute these cues from an image, let me ask you this: without even looking at the image, can we say something about where the sun is? In other words, is there some kind of prior we could use when the cues from the image are uncertain?

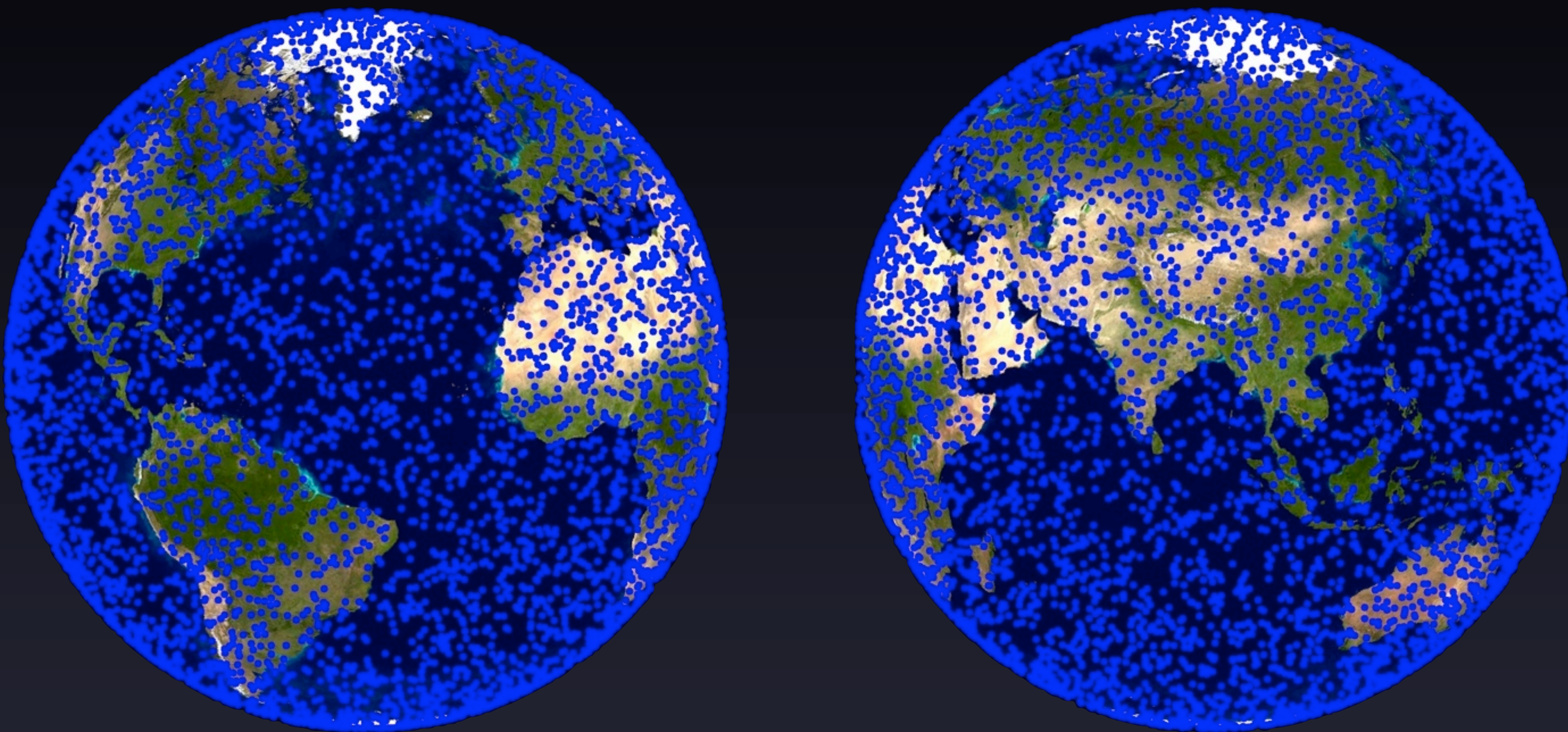
?

[Lalonde, Efros, and Narasimhan, ICCV 2009]

But before we talk about how we compute these cues from an image, let me ask you this: without even looking at the image, can we say something about where the sun is? In other words, is there some kind of prior we could use when the cues from the image are uncertain?

Sun prior

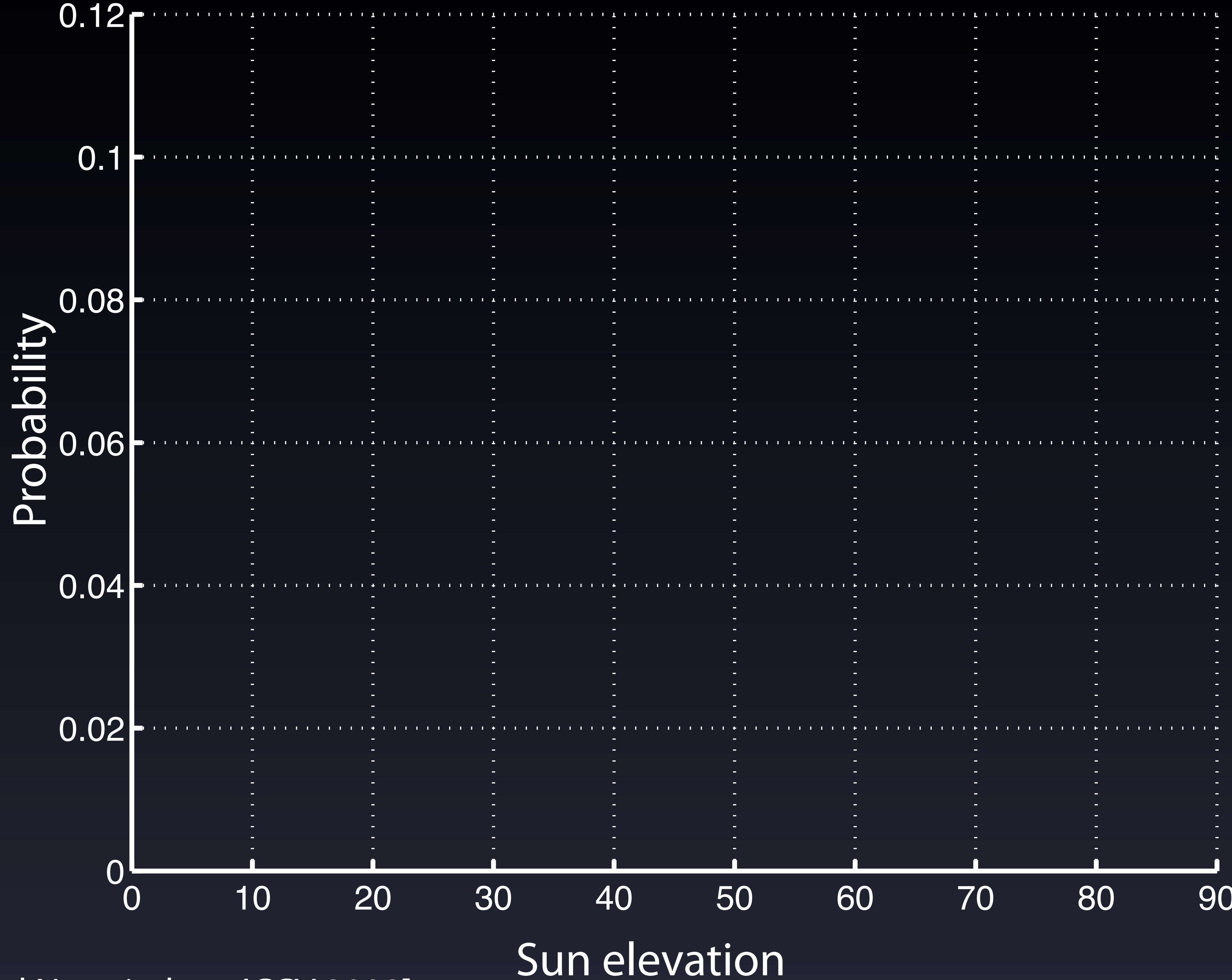
Uniform sampling



[Lalonde, Efros, and Narasimhan, ICCV 2009]

We tried 2 different ideas. First, what if images were captured randomly both in location and time? Well if that was the case, and if we only focus on the sun elevation, we get a probability curve that looks like that.

Sun prior

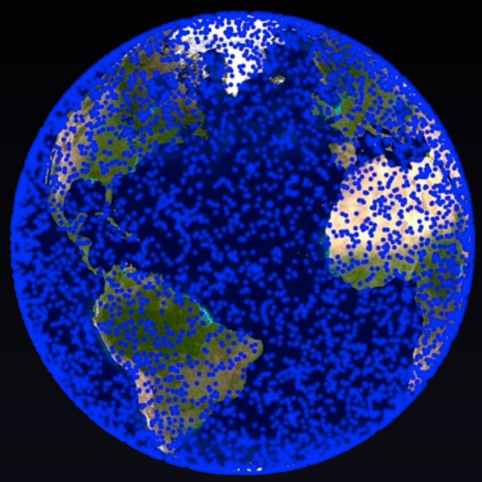


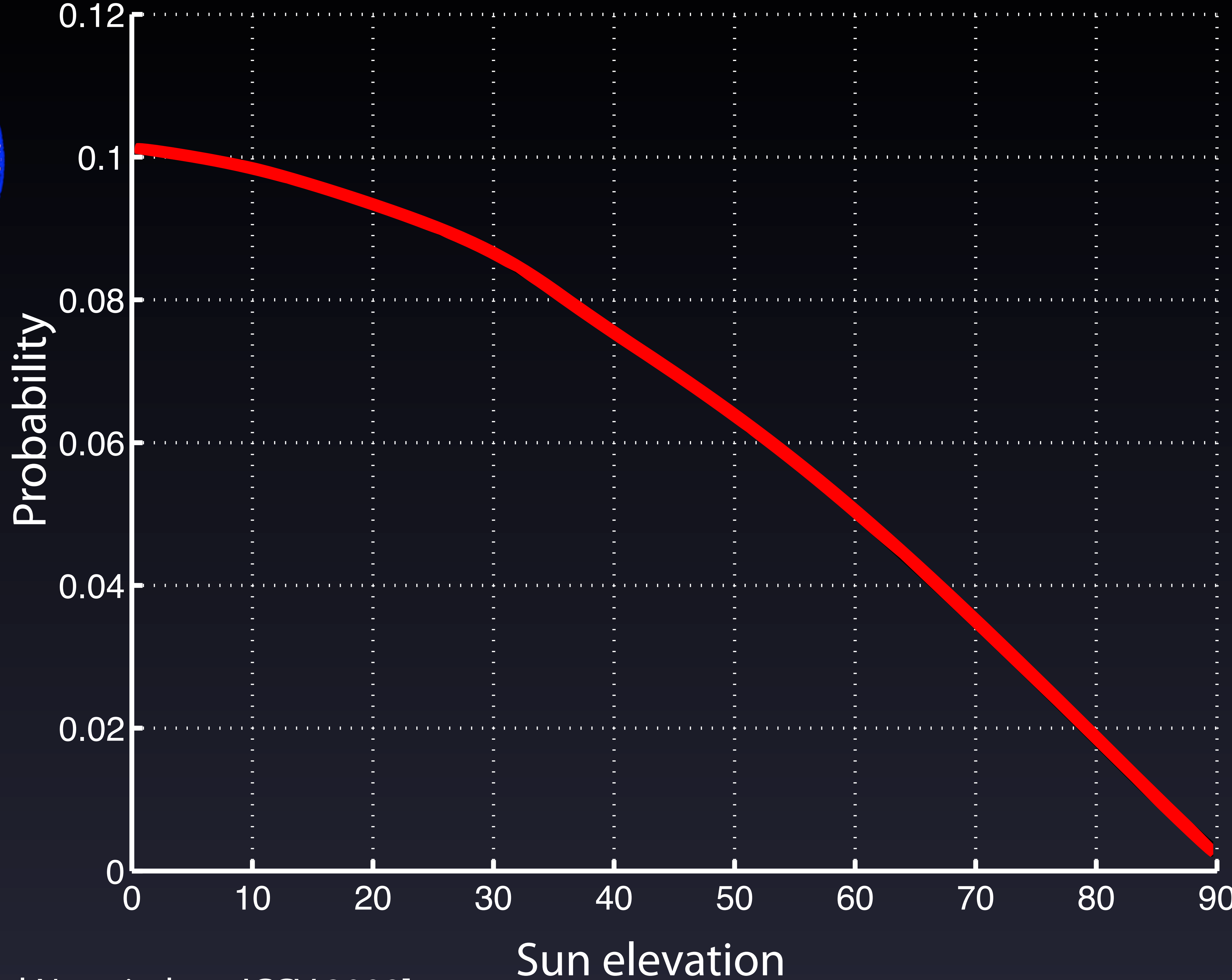
[Lalonde, Efros, and Narasimhan, ICCV 2009]

n the x-axis, we have the sun elevation, in degrees, which ranges from 0 (straight up) to 90 degrees (at the horizon).

Sun prior

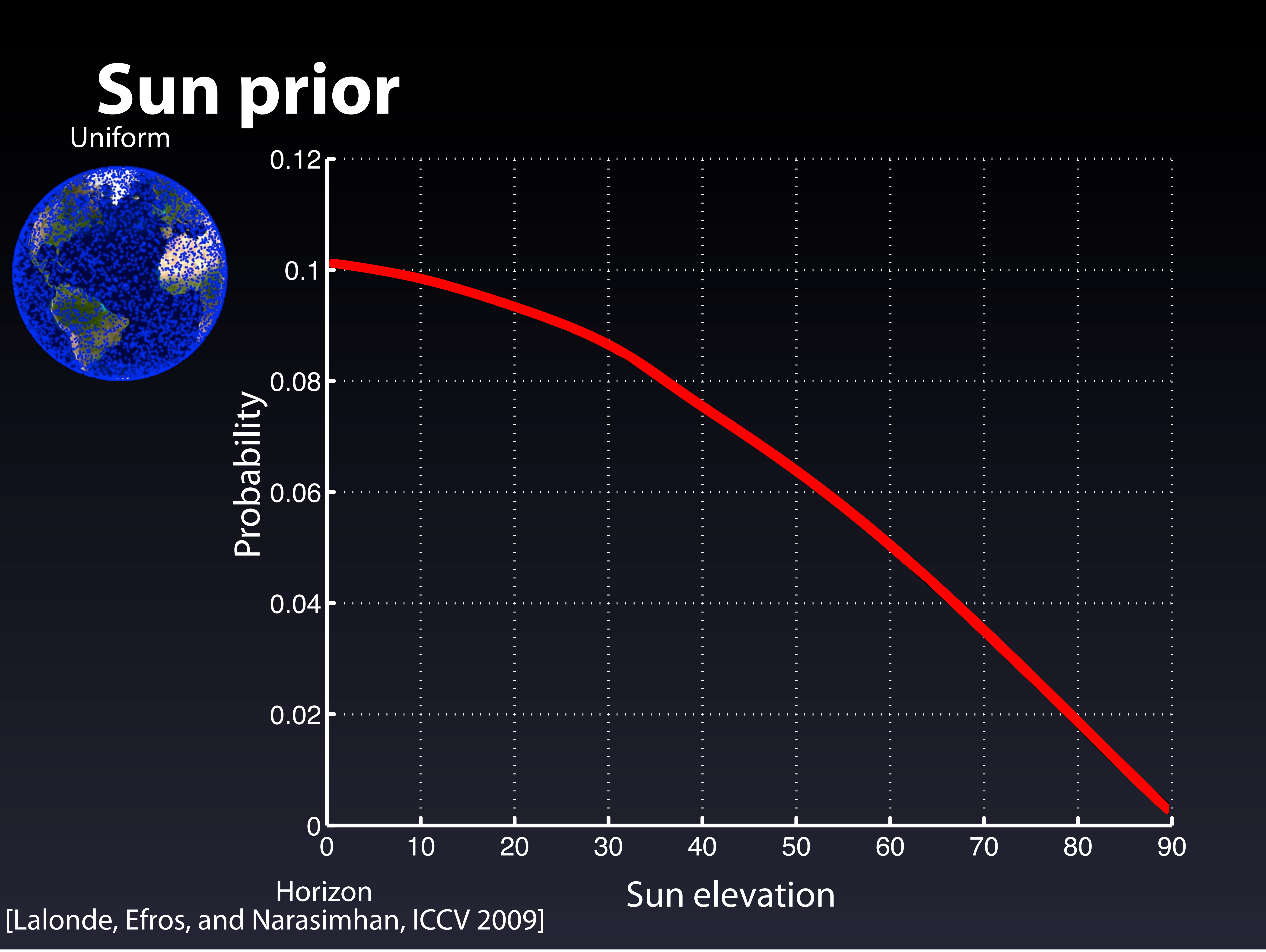
Uniform



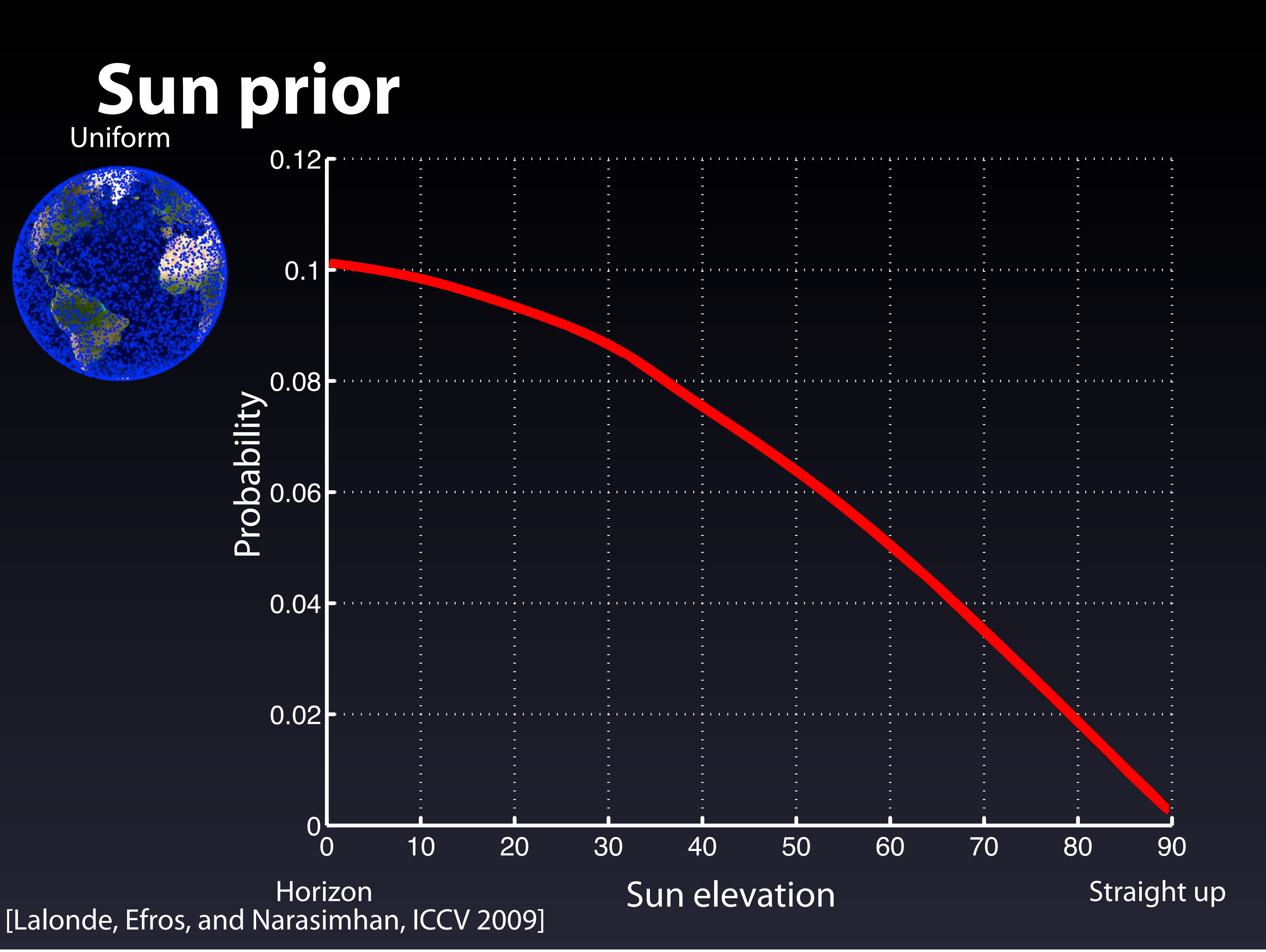


[Lalonde, Efros, and Narasimhan, ICCV 2009]

n the x-axis, we have the sun elevation, in degrees, which ranges from 0 (straight up) to 90 degrees (at the horizon).



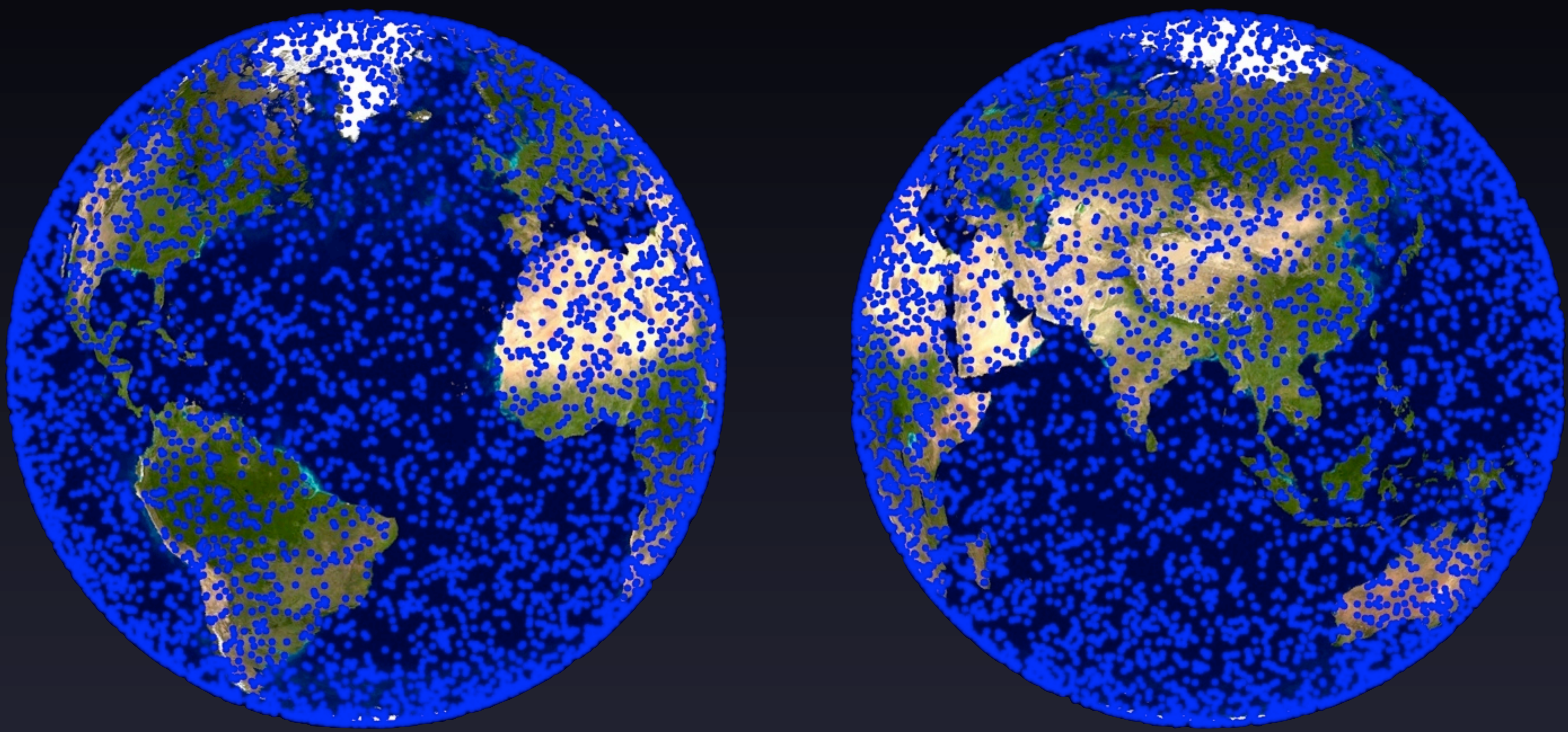
n the x-axis, we have the sun elevation, in degrees, which ranges from 0 (straight up) to 90 degrees (at the horizon).



n the x-axis, we have the sun elevation, in degrees, which ranges from 0 (straight up) to 90 degrees (at the horizon).

Sun prior

Uniform sampling

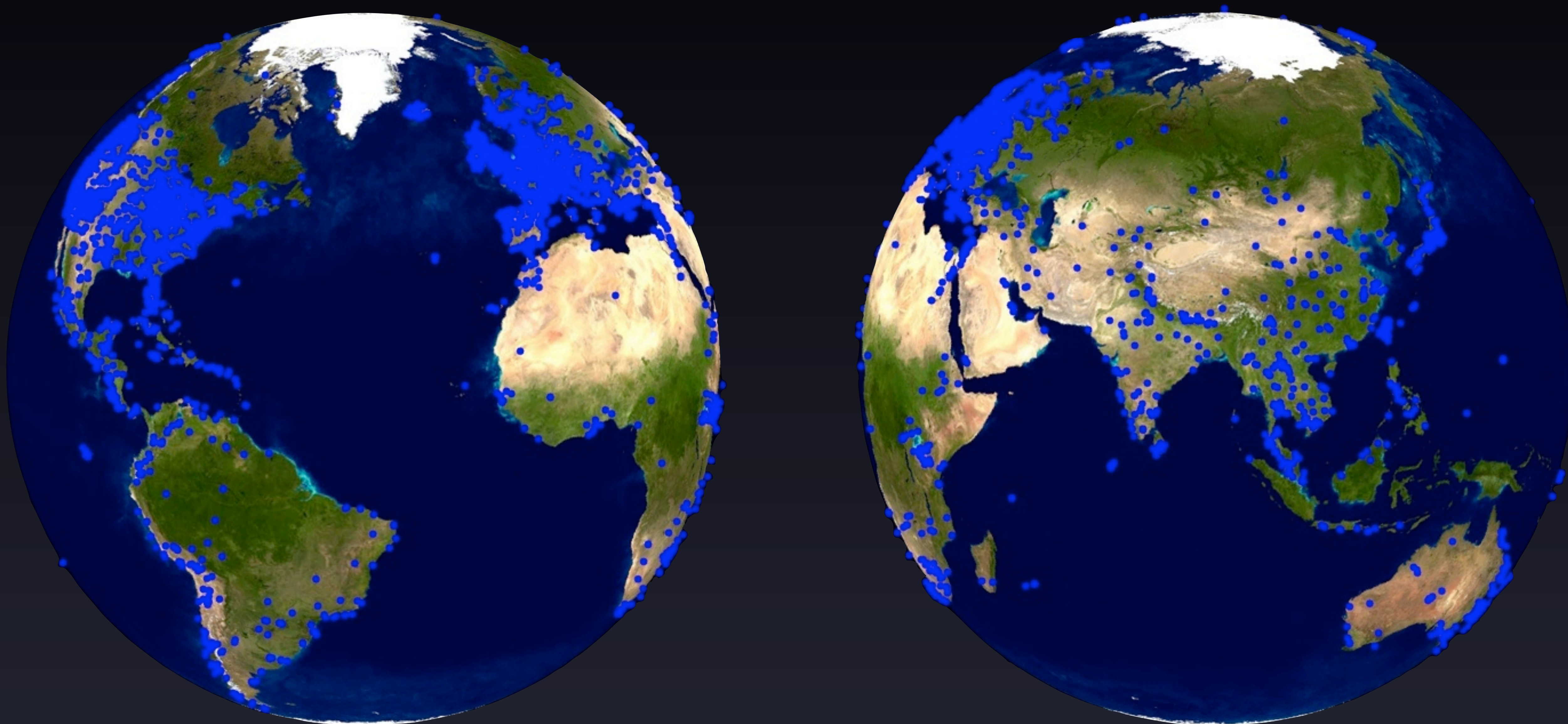


[Lalonde, Efros, and Narasimhan, ICCV 2009]

Of course, people don't take pictures uniformly across the earth, but do so according to a distribution that looks like this one, taken from the 6 million image database of Hays and Efros. If instead we sample the Earth according to that distribution, then the probability curve changes to this.

Sun prior

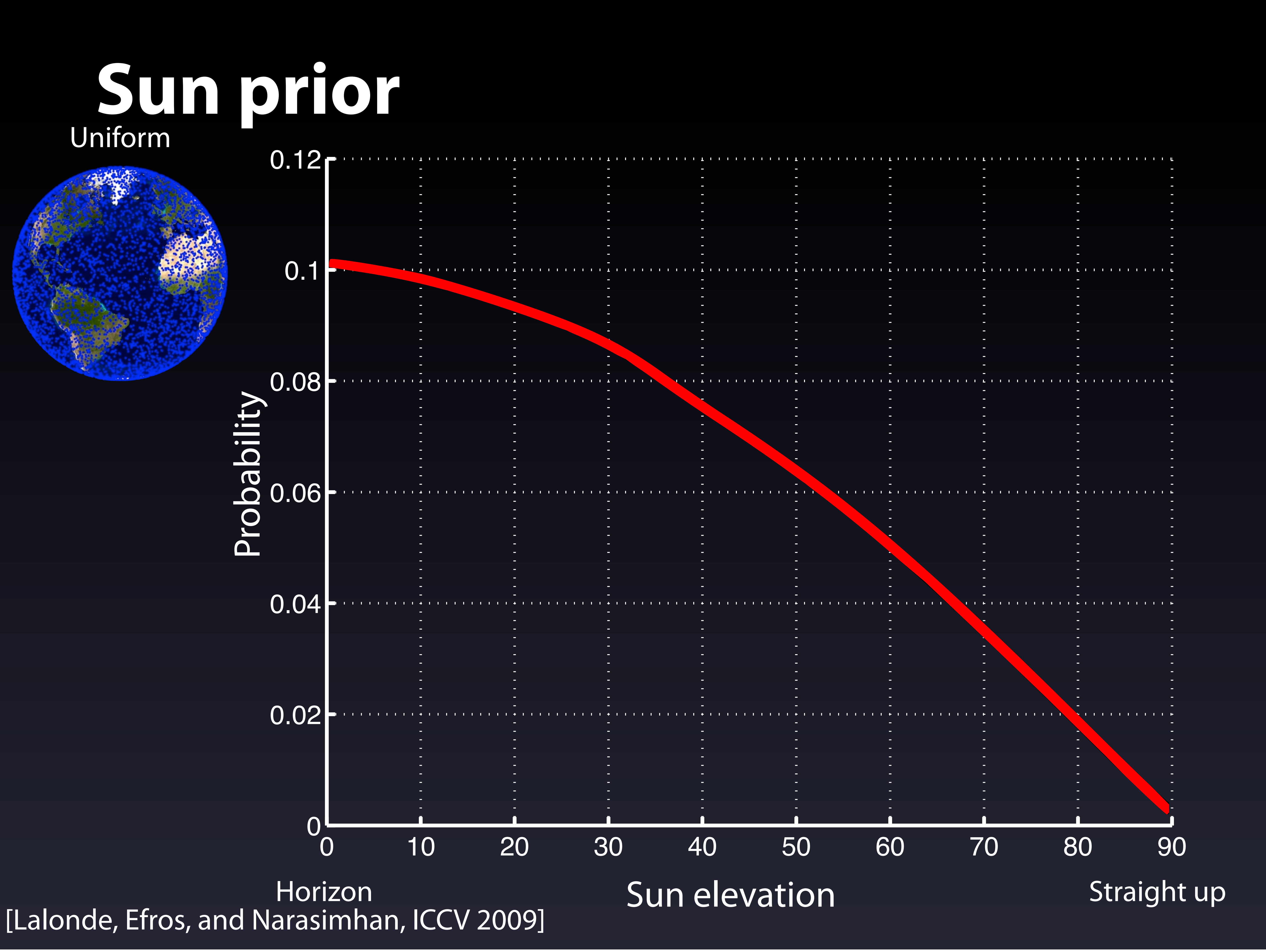
Data-driven sampling (6 million images)



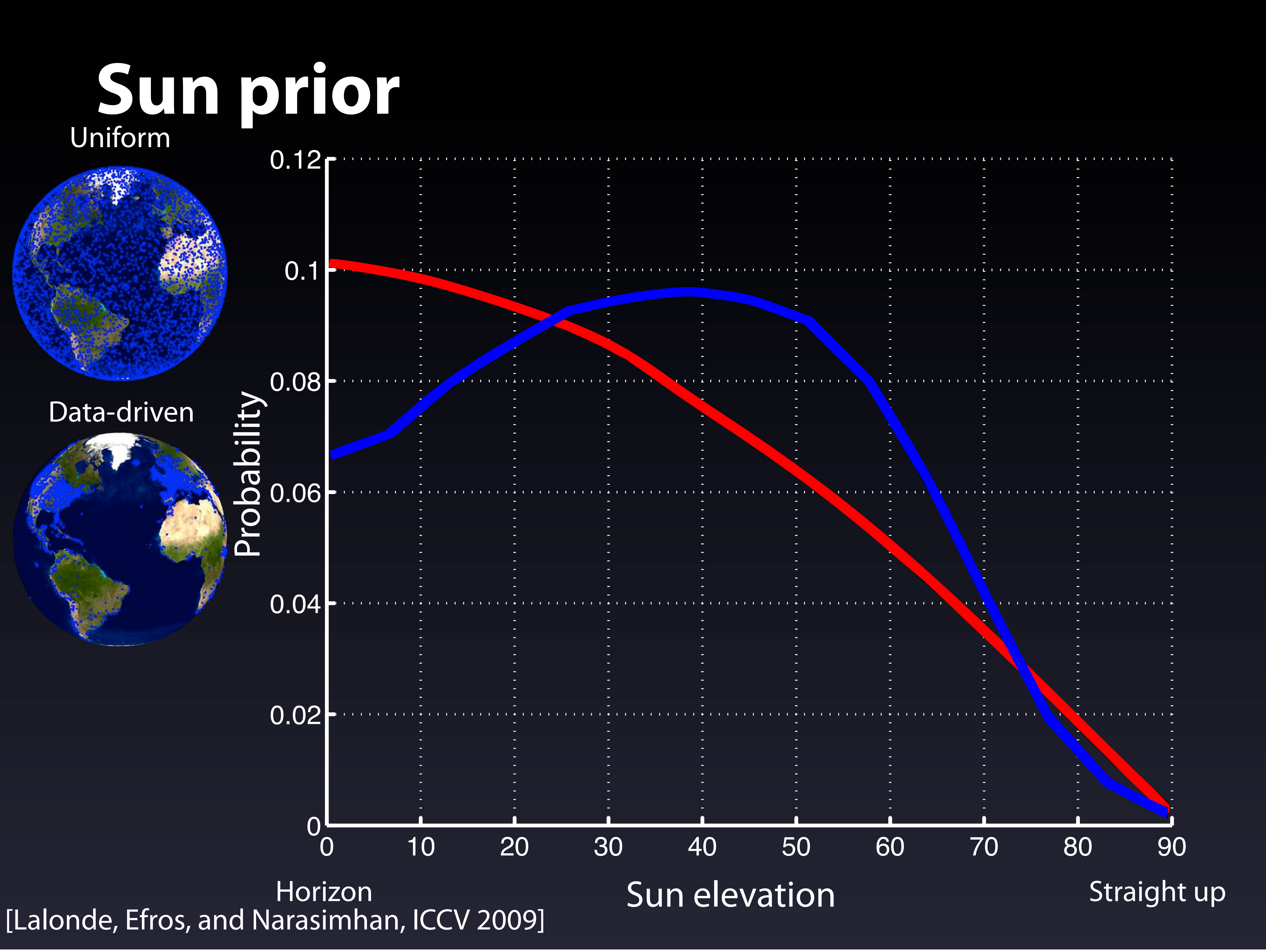
[Lalonde, Efros, and Narasimhan, ICCV 2009]

[Hays and Efros, CVPR '08]

Of course, people don't take pictures uniformly across the earth, but do so according to a distribution that looks like this one, taken from the 6 million image database of Hays and Efros. If instead we sample the Earth according to that distribution, then the probability curve changes to this.



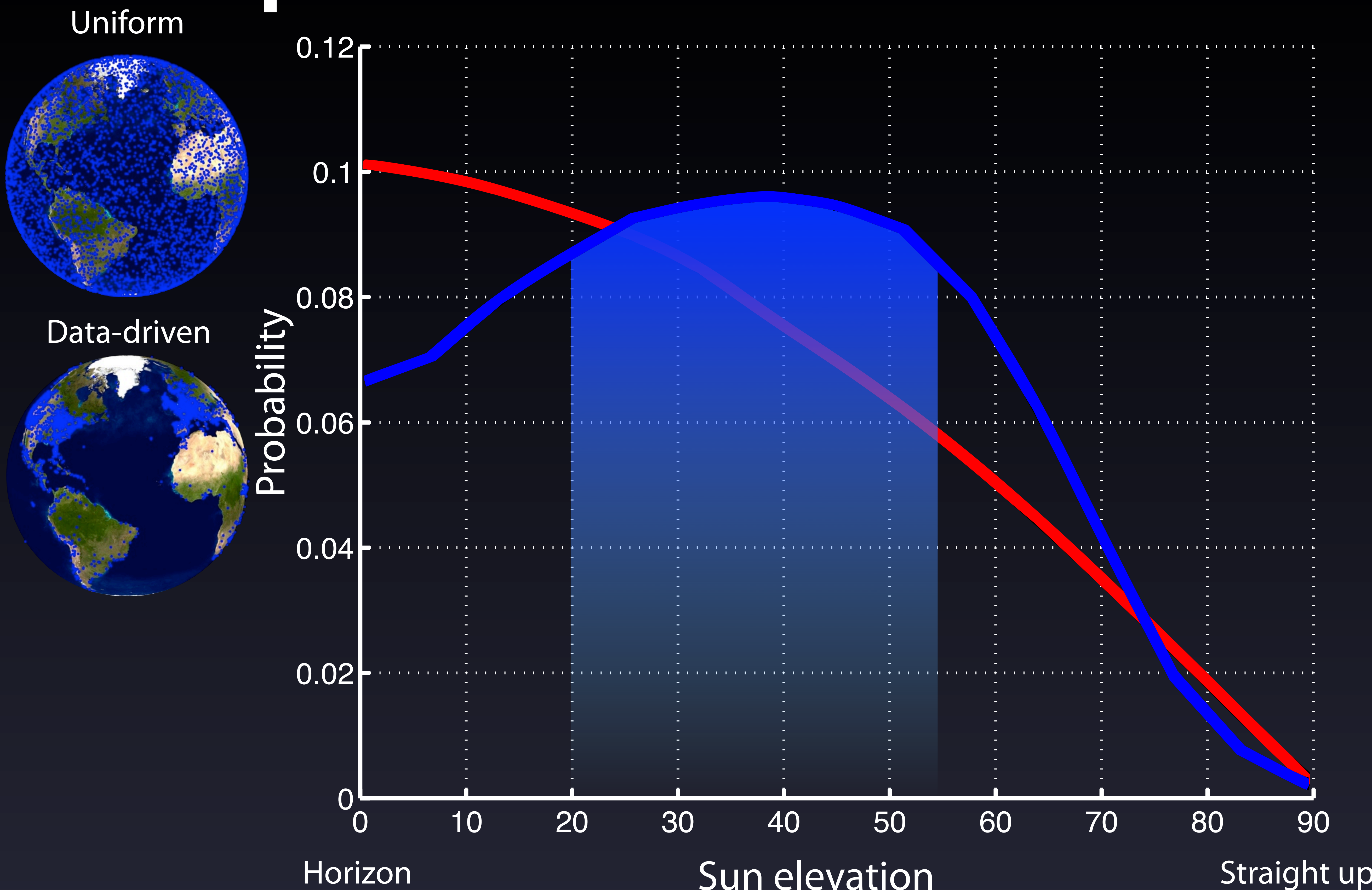
Note the peak between 20–55 degrees which tells us that when people take pictures, it’s more likely that the sun is in this area..
Note that we don’t have azimuth information because that data isn’t available online yet. With the advent of devices with digital compasses like the iPhone, that data should become soon available to get an even more informative prior.



Note the peak between 20-55 degrees which tells us that when people take pictures, it's more likely that the sun is in this area..

Note that we don't have azimuth information because that data isn't available online yet. With the advent of devices with digital compasses like the iPhone, that data should become soon available to get an even more informative prior.

Sun prior

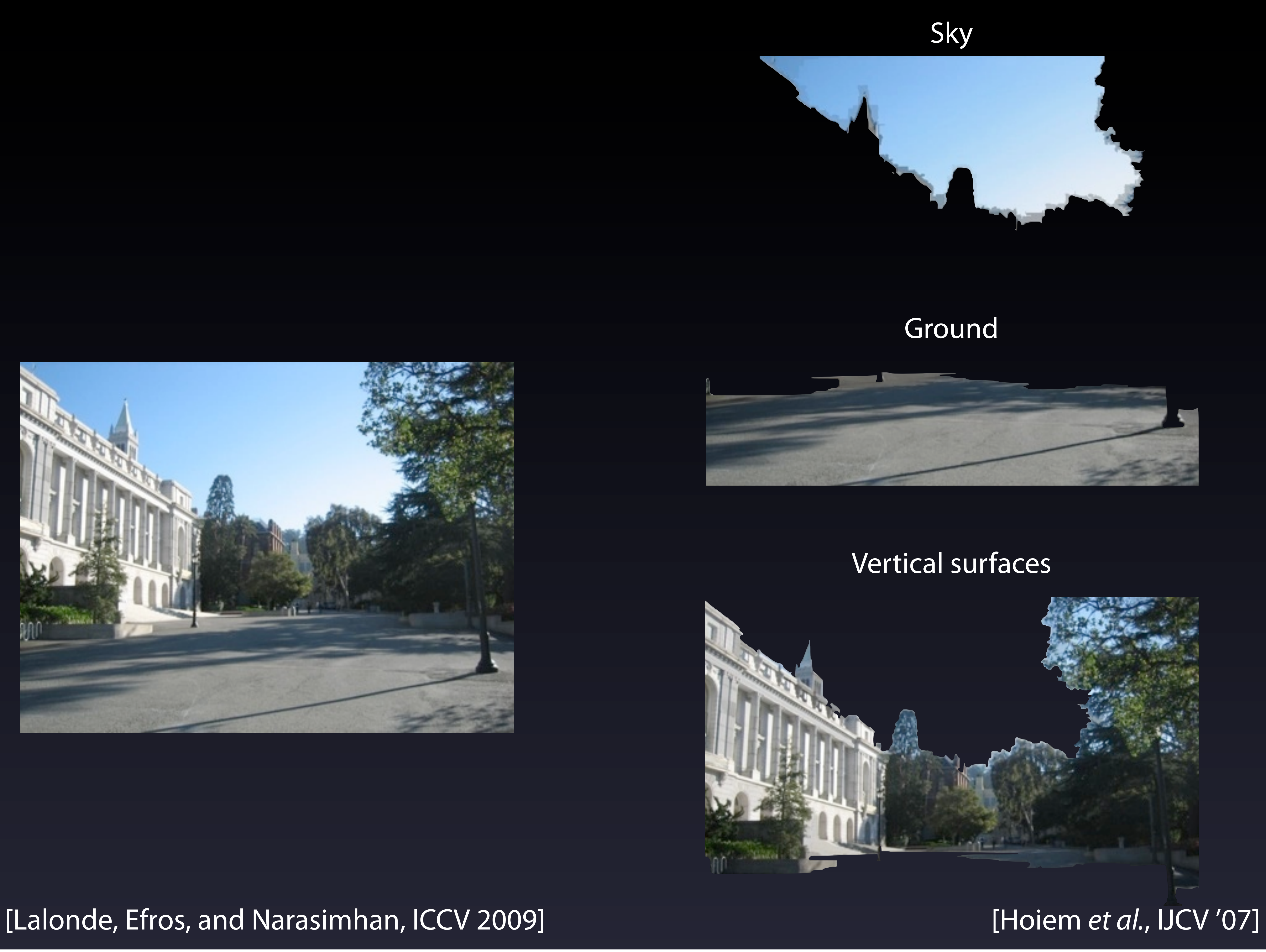


[Lalonde, Efros, and Narasimhan, ICCV 2009]

Note the peak between 20–55 degrees which tells us that when people take pictures, it's more likely that the sun is in this area..
Note that we don't have azimuth information because that data isn't available online yet. With the advent of devices with digital compasses like the iPhone, that data should become soon available to get an even more informative prior.

[Lalonde, Efros, and Narasimhan, ICCV 2009]

But of course we also need to look at the image, so let's see how we compute these cues now. First, we split the image into our 3 regions, using the existing geometric context algorithm of Hoiem et al. and compute the cues on each one of them independently. Let's get started by looking at the sky. What information is really available on the sky region? Is it really just a patch of blue pixels as we too often think?



But of course we also need to look at the image, so let's see how we compute these cues now. First, we split the image into our 3 regions, using the existing geometric context algorithm of Hoiem et al. and compute the cues on each one of them independently. Let's get started by looking at the sky. What information is really available on the sky region? Is it really just a patch of blue pixels as we too often think?

[Lalonde, Efros, and Narasimhan, ICCV 2009]

But of course we also need to look at the image, so let's see how we compute these cues now. First, we split the image into our 3 regions, using the existing geometric context algorithm of Hoiem et al. and compute the cues on each one of them independently. Let's get started by looking at the sky. What information is really available on the sky region? Is it really just a patch of blue pixels as we too often think?

Illumination from webcams

[Lalonde, Narasimhan & Efros, ECCV ’08 + IJCV ’09]

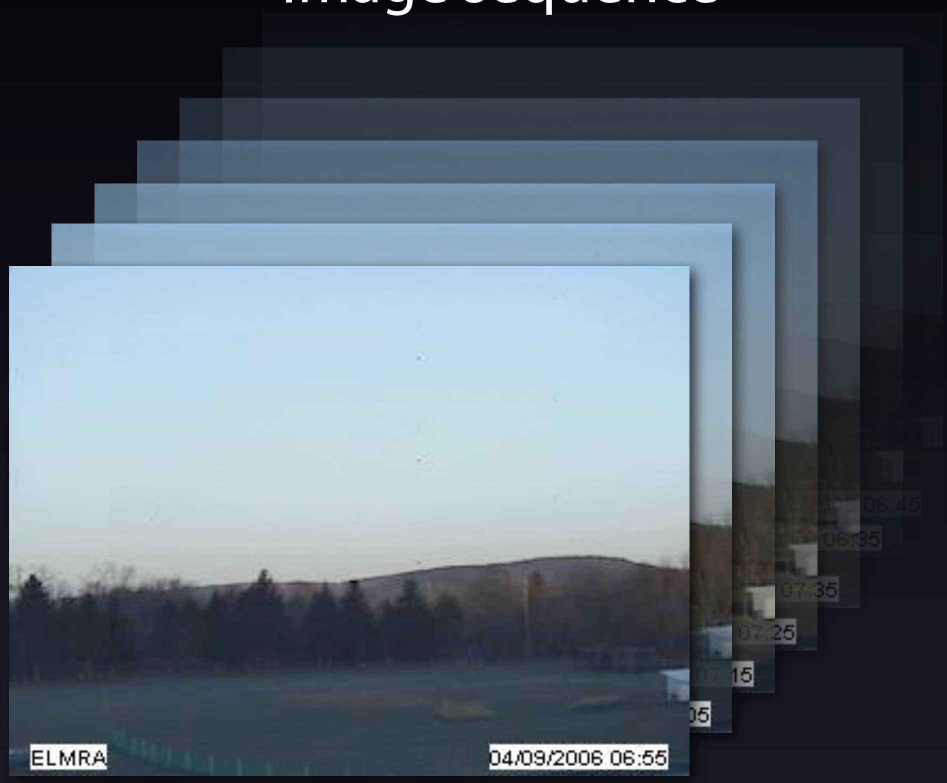
[Lalonde, Efros, and Narasimhan, ICCV 2009]

[Perez *et al.*, ’93]

In ECCV last year, we showed that in image sequences such as a webcam, the clear sky alone can be used to recover 3 important camera parameters: its zenith angle, azimuth angle, and focal length. For this we used a physically-based model of the sky appearance from Perez et al. Once this is known, we can recover at every frame in the sequence, the sun position, the sky color everywhere, and even a cloud segmentation. In short, the sky alone can be used to estimate the natural illumination parameters of each frame in an image sequence.

Illumination from webcams

Image sequence



[Lalonde, Narasimhan & Efros, ECCV ’08 + IJCV ’09]

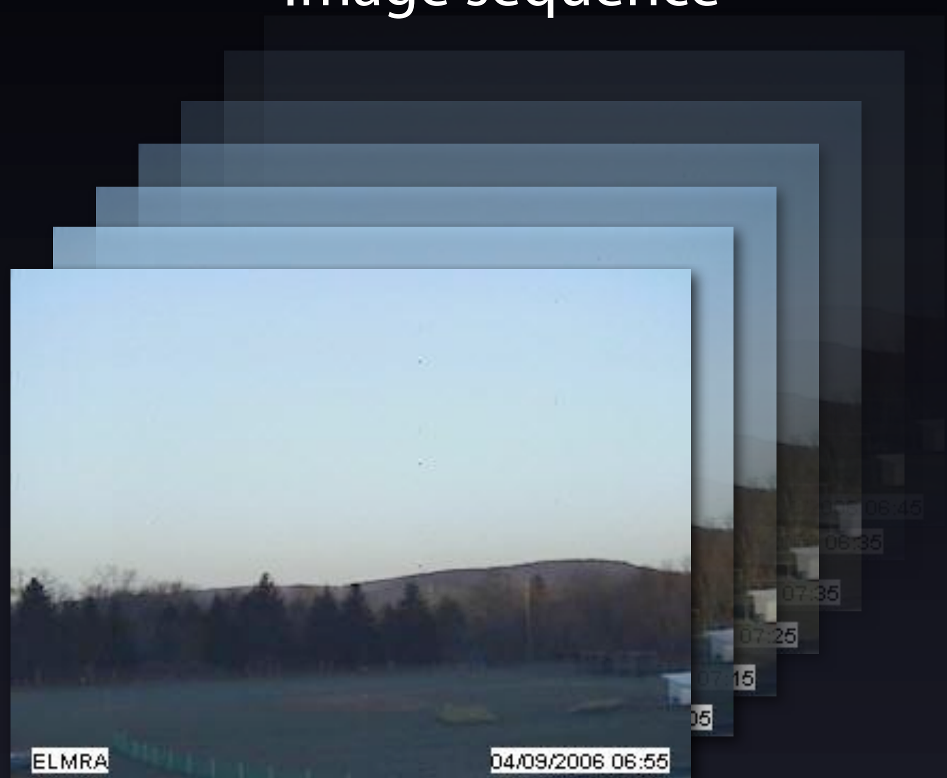
[Lalonde, Efros, and Narasimhan, ICCV 2009]

[Perez *et al.*, ’93]

In ECCV last year, we showed that in image sequences such as a webcam, the clear sky alone can be used to recover 3 important camera parameters: its zenith angle, azimuth angle, and focal length. For this we used a physically-based model of the sky appearance from Perez et al. Once this is known, we can recover at every frame in the sequence, the sun position, the sky color everywhere, and even a cloud segmentation. In short, the sky alone can be used to estimate the natural illumination parameters of each frame in an image sequence.

Illumination from webcams

Image sequence



[Lalonde, Narasimhan & Efros, ECCV ’08 + IJCV ’09]
[Perez *et al.*, ’93]

[Lalonde, Efros, and Narasimhan, ICCV 2009]

In ECCV last year, we showed that in image sequences such as a webcam, the clear sky alone can be used to recover 3 important camera parameters: its zenith angle, azimuth angle, and focal length. For this we used a physically-based model of the sky appearance from Perez et al. Once this is known, we can recover at every frame in the sequence, the sun position, the sky color everywhere, and even a cloud segmentation. In short, the sky alone can be used to estimate the natural illumination parameters of each frame in an image sequence.

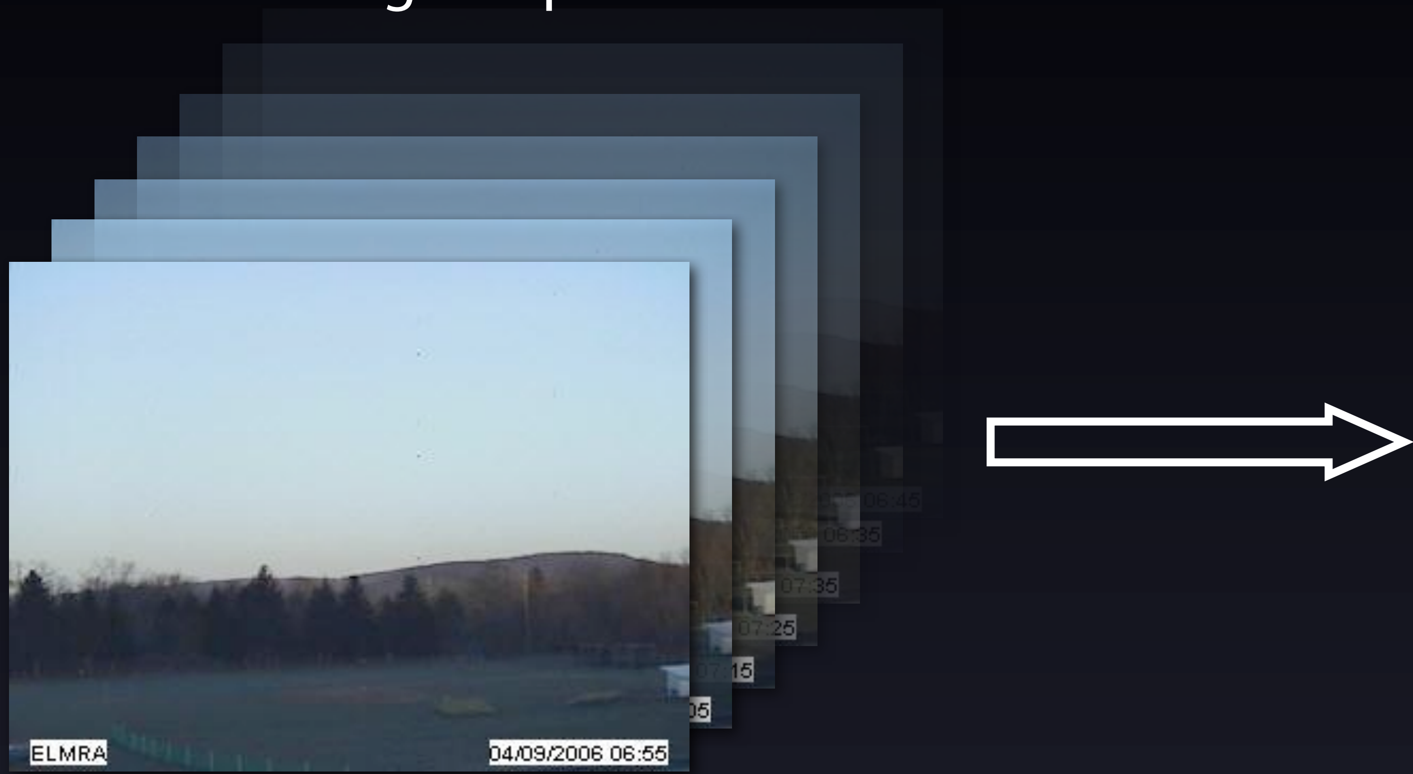
Illumination from webcams

[Lalonde, Narasimhan & Efros, ECCV '08 + IJCV '09]
[Perez *et al.*, '93]
[Lalonde, Efros, and Narasimhan, ICCV 2009]

In ECCV last year, we showed that in image sequences such as a webcam, the clear sky alone can be used to recover 3 important camera parameters: its zenith angle, azimuth angle, and focal length. For this we used a physically-based model of the sky appearance from Perez et al. Once this is known, we can recover at every frame in the sequence, the sun position, the sky color everywhere, and even a cloud segmentation. In short, the sky alone can be used to estimate the natural illumination parameters of each frame in an image sequence.

Illumination from webcams

Image sequence

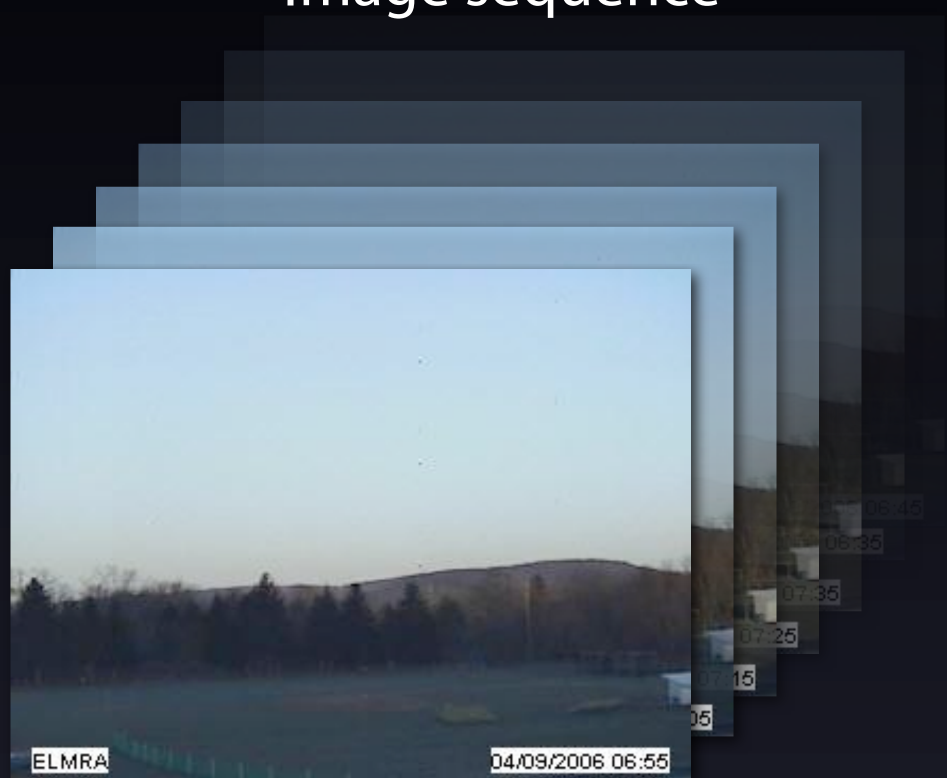


[Lalonde, Narasimhan & Efros, ECCV ’08 + IJCV ’09]
[Lalonde, Efros, and Narasimhan, ICCV 2009]
[Perez *et al.*, ’93]

In ECCV last year, we showed that in image sequences such as a webcam, the clear sky alone can be used to recover 3 important camera parameters: its zenith angle, azimuth angle, and focal length. For this we used a physically-based model of the sky appearance from Perez et al. Once this is known, we can recover at every frame in the sequence, the sun position, the sky color everywhere, and even a cloud segmentation. In short, the sky alone can be used to estimate the natural illumination parameters of each frame in an image sequence.

Illumination from webcams

Image sequence



+ zoom (f)

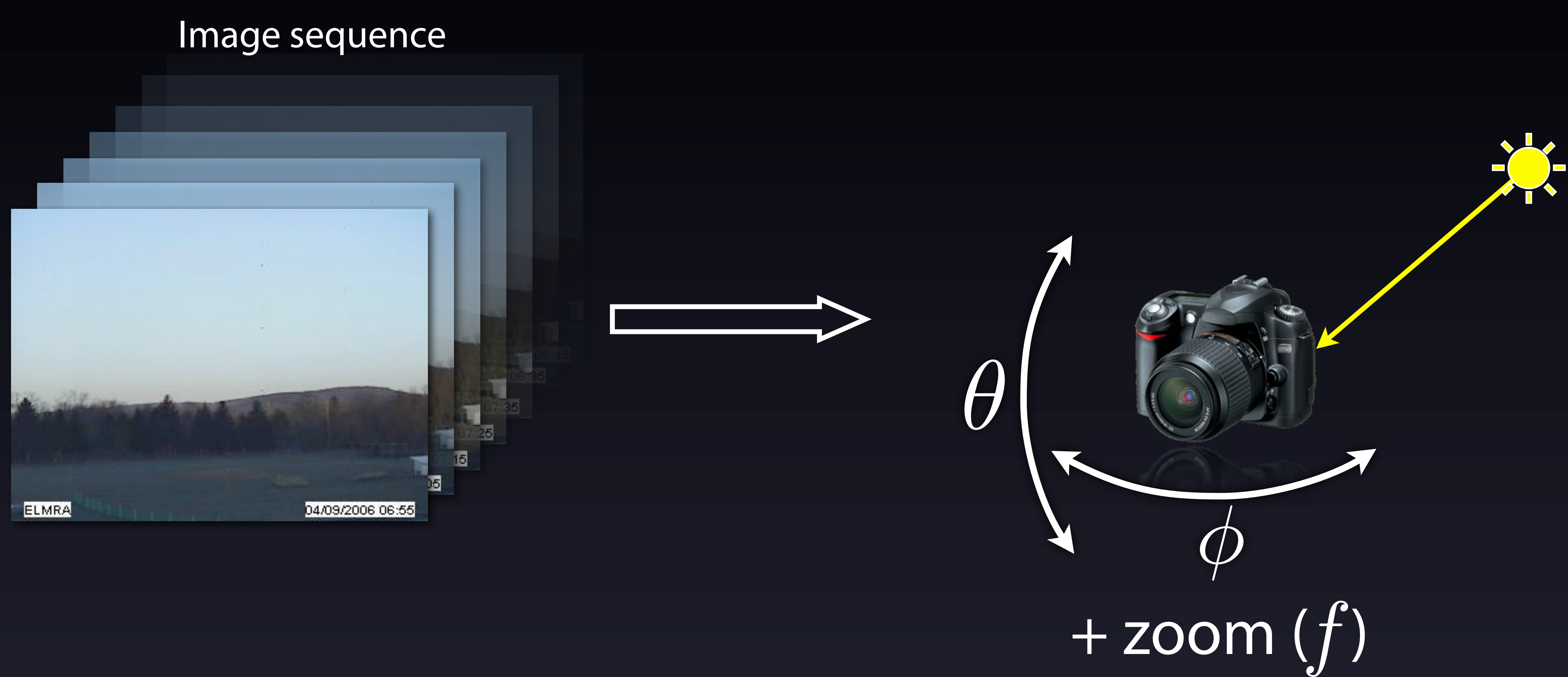
[Lalonde, Narasimhan & Efros, ECCV '08 + IJCV '09]

[Perez *et al.*, '93]

[Lalonde, Efros, and Narasimhan, ICCV 2009]

In ECCV last year, we showed that in image sequences such as a webcam, the clear sky alone can be used to recover 3 important camera parameters: its zenith angle, azimuth angle, and focal length. For this we used a physically-based model of the sky appearance from Perez et al. Once this is known, we can recover at every frame in the sequence, the sun position, the sky color everywhere, and even a cloud segmentation. In short, the sky alone can be used to estimate the natural illumination parameters of each frame in an image sequence.

Illumination from webcams



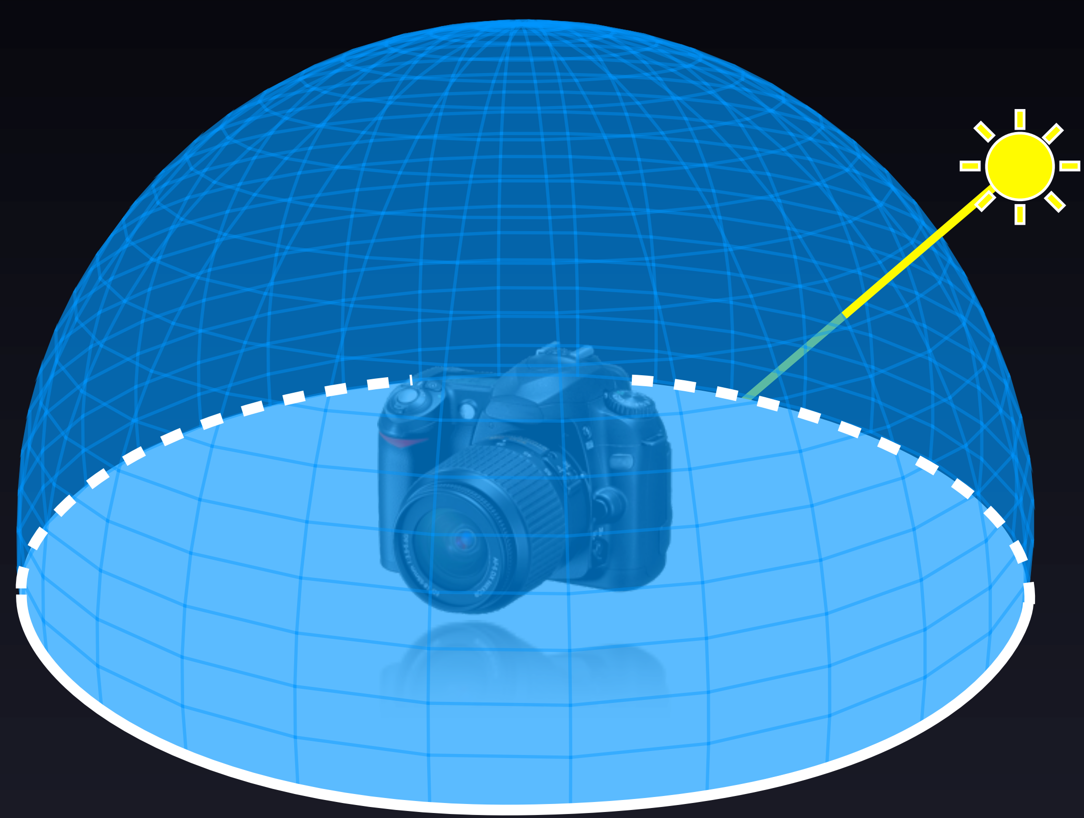
[Lalonde, Narasimhan & Efros, ECCV '08 + IJCV '09]

[Lalonde, Efros, and Narasimhan, ICCV 2009]

[Perez *et al.*, '93]

In ECCV last year, we showed that in image sequences such as a webcam, the clear sky alone can be used to recover 3 important camera parameters: its zenith angle, azimuth angle, and focal length. For this we used a physically-based model of the sky appearance from Perez et al. Once this is known, we can recover at every frame in the sequence, the sun position, the sky color everywhere, and even a cloud segmentation. In short, the sky alone can be used to estimate the natural illumination parameters of each frame in an image sequence.

Illumination from webcams



[Lalonde, Narasimhan & Efros, ECCV '08 + IJCV '09]

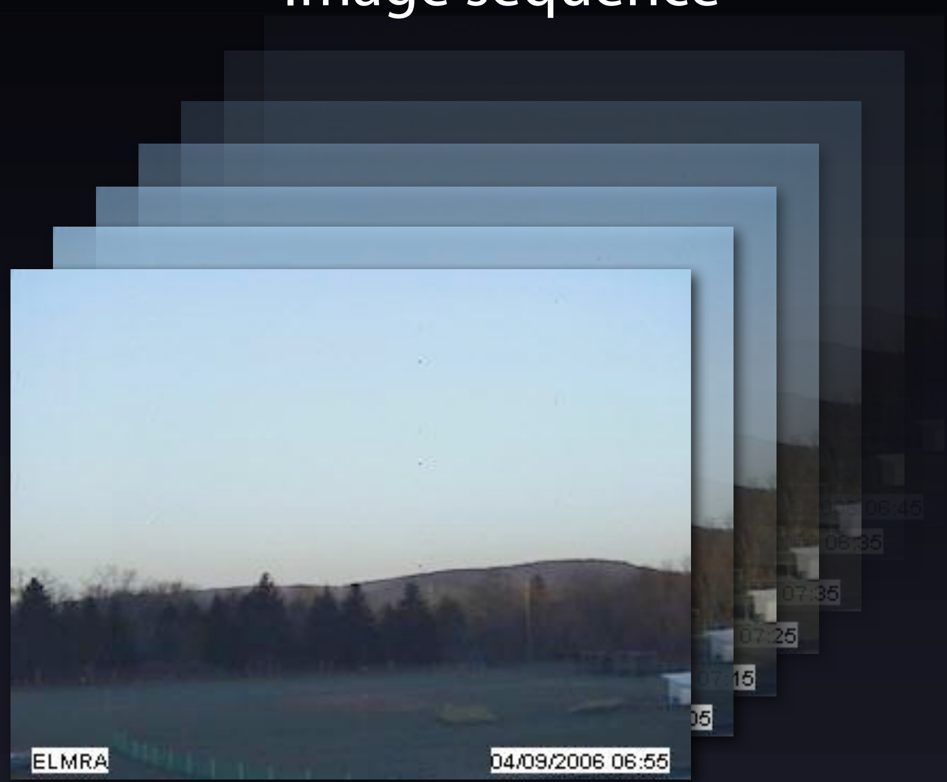
[Lalonde, Efros, and Narasimhan, ICCV 2009]

[Perez *et al.*, '93]

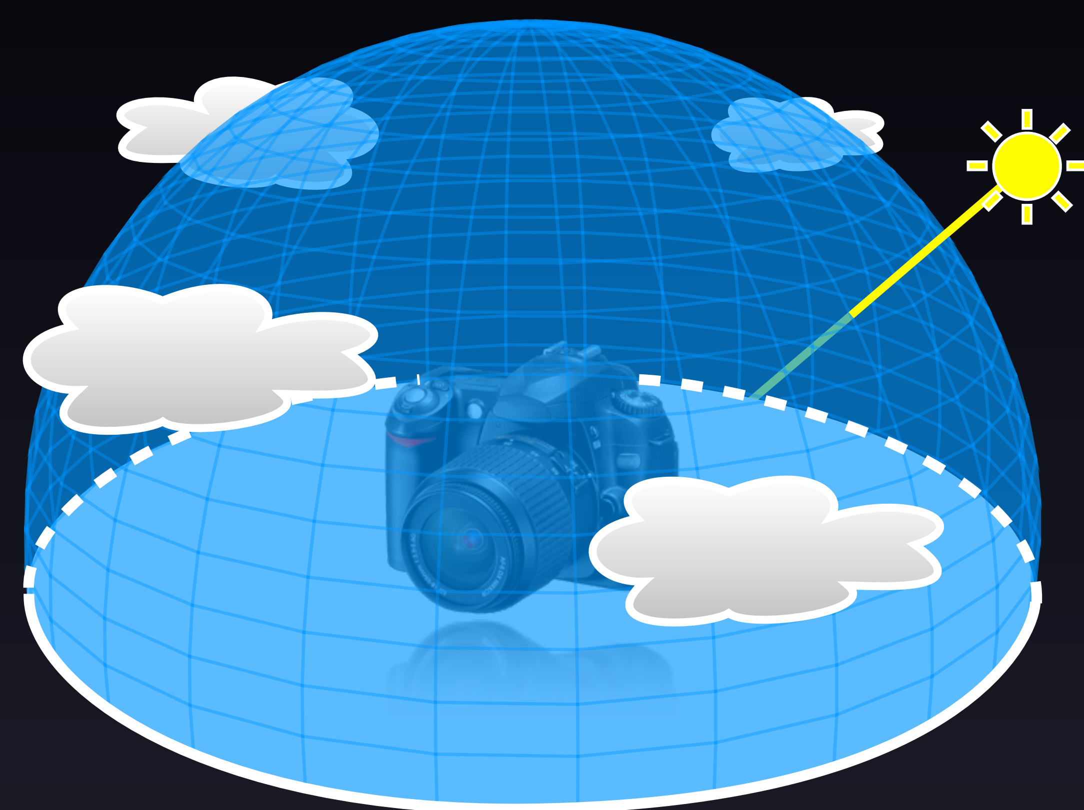
In ECCV last year, we showed that in image sequences such as a webcam, the clear sky alone can be used to recover 3 important camera parameters: its zenith angle, azimuth angle, and focal length. For this we used a physically-based model of the sky appearance from Perez et al. Once this is known, we can recover at every frame in the sequence, the sun position, the sky color everywhere, and even a cloud segmentation. In short, the sky alone can be used to estimate the natural illumination parameters of each frame in an image sequence.

Illumination from webcams

Image sequence



Natural illumination



[Lalonde, Narasimhan & Efros, ECCV '08 + IJCV '09]

[Lalonde, Efros, and Narasimhan, ICCV 2009]

[Perez *et al.*, '93]

In ECCV last year, we showed that in image sequences such as a webcam, the clear sky alone can be used to recover 3 important camera parameters: its zenith angle, azimuth angle, and focal length. For this we used a physically-based model of the sky appearance from Perez et al. Once this is known, we can recover at every frame in the sequence, the sun position, the sky color everywhere, and even a cloud segmentation. In short, the sky alone can be used to estimate the natural illumination parameters of each frame in an image sequence.

Webcams vs single images

[Lalonde, Efros, and Narasimhan, ICCV 2009]

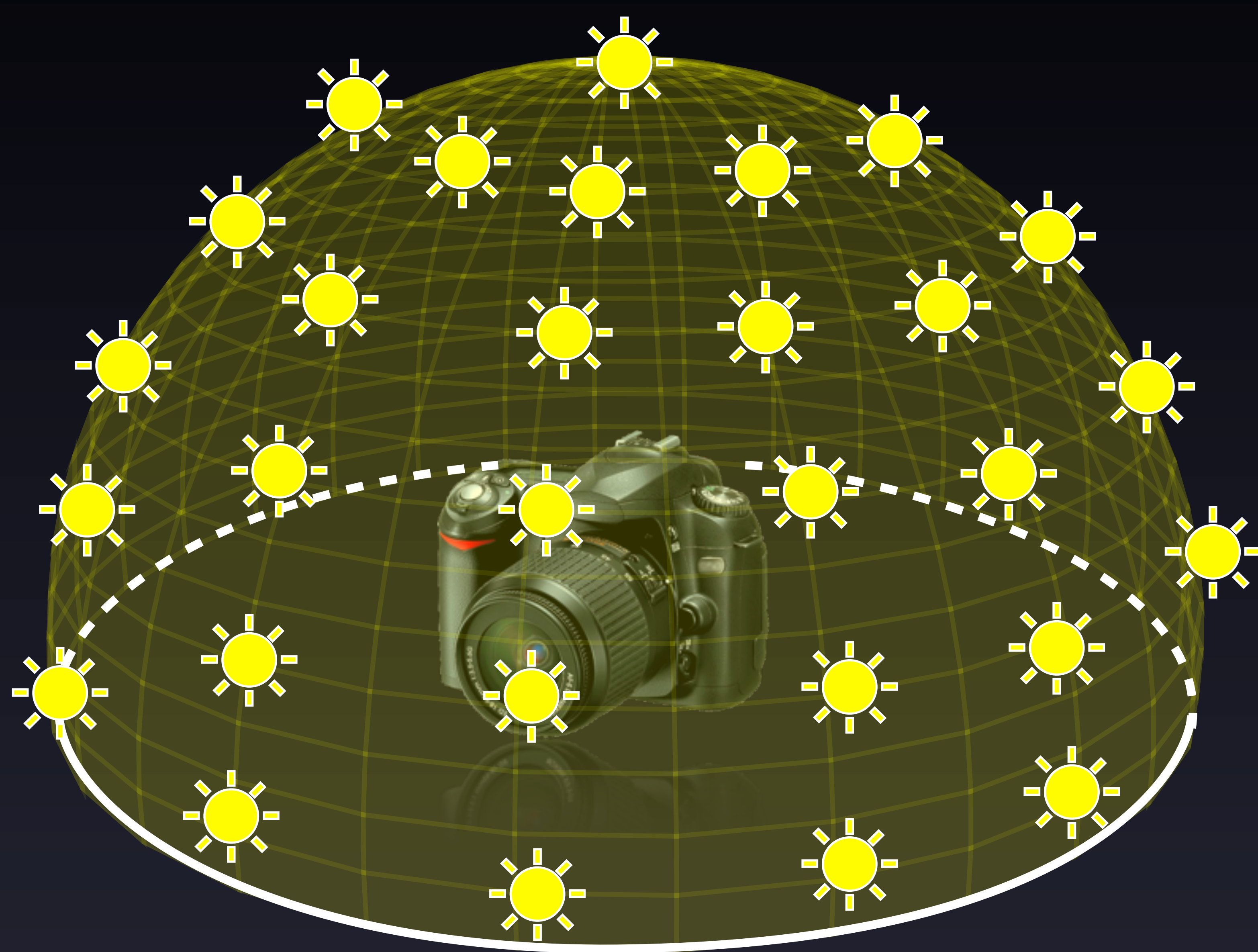
Now, we only have a single image, so things are much more under-constrained.
Instead of trying to find the most likely camera parameters, or equivalently, relative sun position, as we did in the previous work, we'll try to find the distribution over all sun positions with respect to the camera. In practice, we discretize the elevation-azimuth space and estimate the probability of the sun being at each location. And to represent this distribution on sun positions,

Webcams vs single images

[Lalonde, Efros, and Narasimhan, ICCV 2009]

Now, we only have a single image, so things are much more under-constrained.
Instead of trying to find the most likely camera parameters, or equivalently, relative sun position, as we did in the previous work, we'll try to find the distribution over all sun positions with respect to the camera. In practice, we discretize the elevation-azimuth space and estimate the probability of the sun being at each location. And to represent this distribution on sun positions,

Webcams vs single images



[Lalonde, Efros, and Narasimhan, ICCV 2009]

Now, we only have a single image, so things are much more under-constrained.
Instead of trying to find the most likely camera parameters, or equivalently, relative sun position, as we did in the previous work, we'll try to find the distribution over all sun positions with respect to the camera. In practice, we discretize the elevation-azimuth space and estimate the probability of the sun being at each location. And to represent this distribution on sun positions,

Bottom-up projection

[Lalonde, Efros, and Narasimhan, ICCV 2009]

we'll employ this bottom up projection, where
- the central point is straight above our head and the surrounding circle is the horizon.
- below is facing forward, with the corresponding camera field of view.
- and accordingly, we have right, back and left.

We display the probability of the sun being at each location using colors ranging from not probably to high probable.

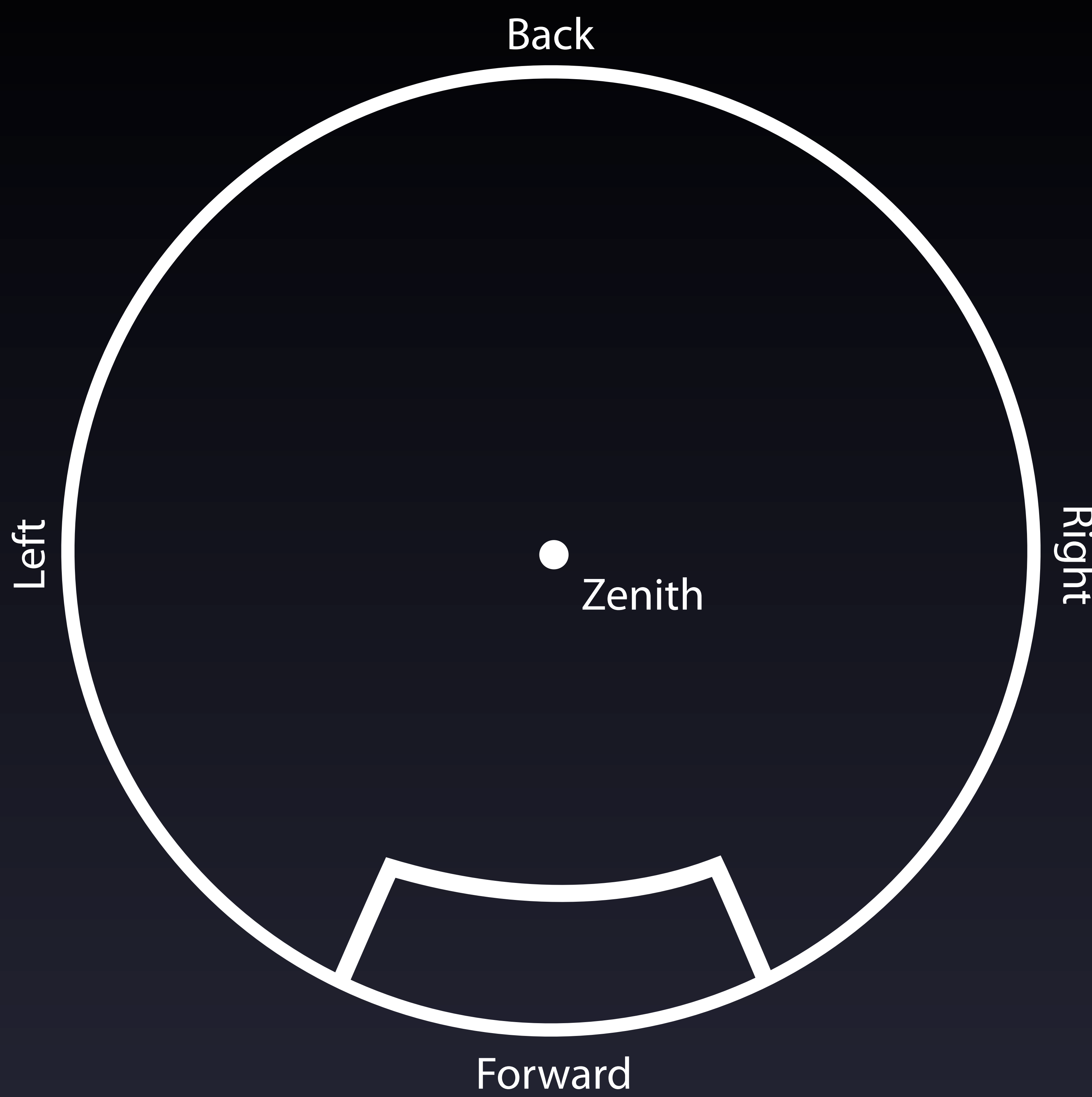
Bottom-up projection

[Lalonde, Efros, and Narasimhan, ICCV 2009]

we'll employ this bottom up projection, where
- the central point is straight above our head and the surrounding circle is the horizon.
- below is facing forward, with the corresponding camera field of view.
- and accordingly, we have right, back and left.

We display the probability of the sun being at each location using colors ranging from not probably to high probable.

Bottom-up projection

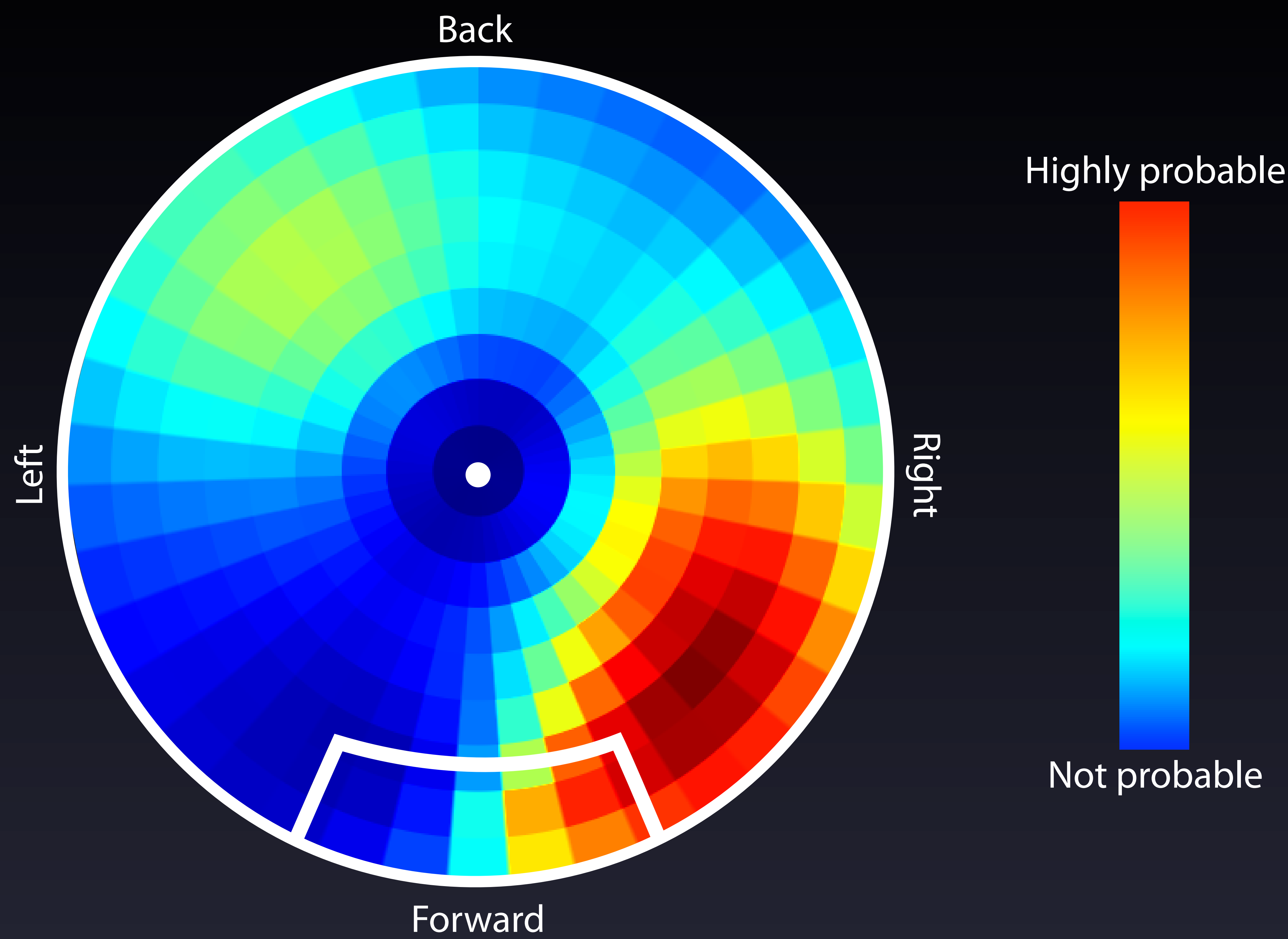


[Lalonde, Efros, and Narasimhan, ICCV 2009]

we'll employ this bottom up projection, where
- the central point is straight above our head and the surrounding circle is the horizon.
- below is facing forward, with the corresponding camera field of view.
- and accordingly, we have right, back and left.

We display the probability of the sun being at each location using colors ranging from not probably to high probable.

Bottom-up projection



[Lalonde, Efros, and Narasimhan, ICCV 2009]

we'll employ this bottom up projection, where
- the central point is straight above our head and the surrounding circle is the horizon.
- below is facing forward, with the corresponding camera field of view.
- and accordingly, we have right, back and left.

We display the probability of the sun being at each location using colors ranging from not probably to high probable.

Sky

[Lalonde, Efros, and Narasimhan, ICCV 2009]

Let’s take an image and see how we can obtain such a probability map from its sky pixels.

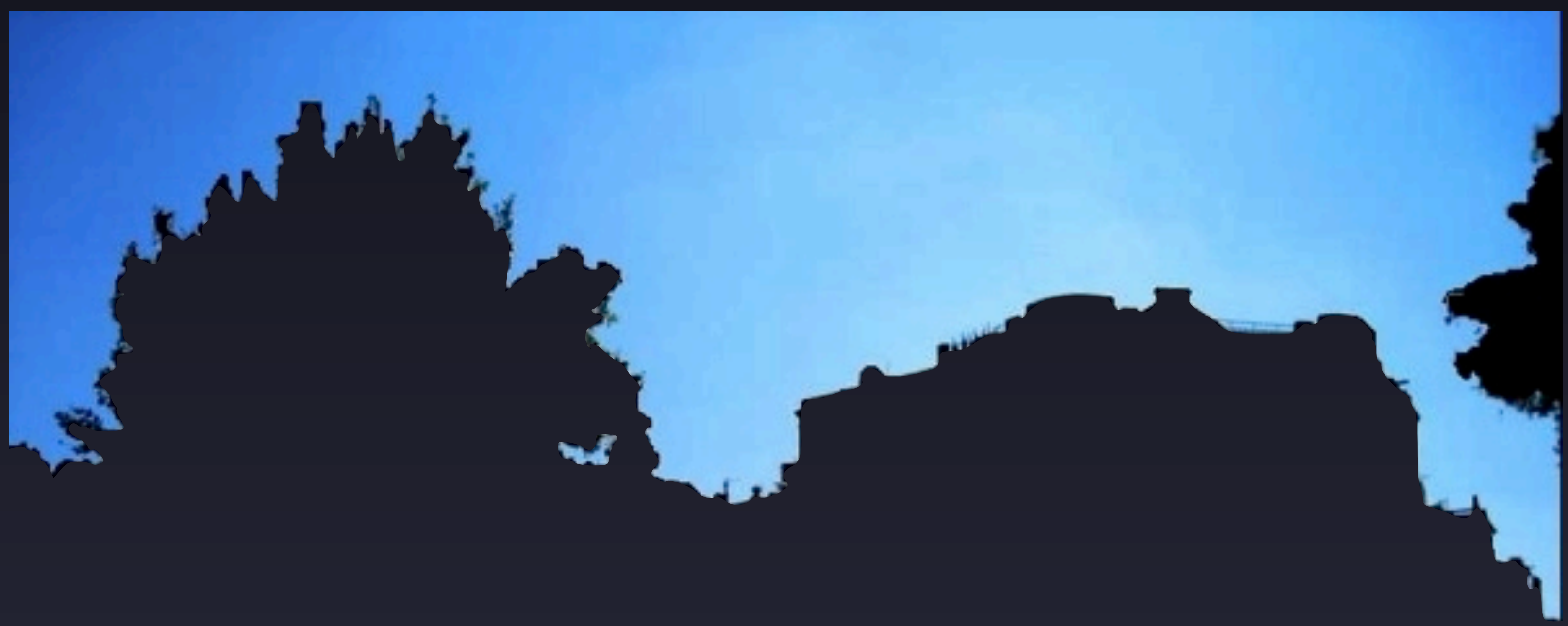
Sky

[Lalonde, Efros, and Narasimhan, ICCV 2009]

Let’s take an image and see how we can obtain such a probability map from its sky pixels.

Sky

Original sky

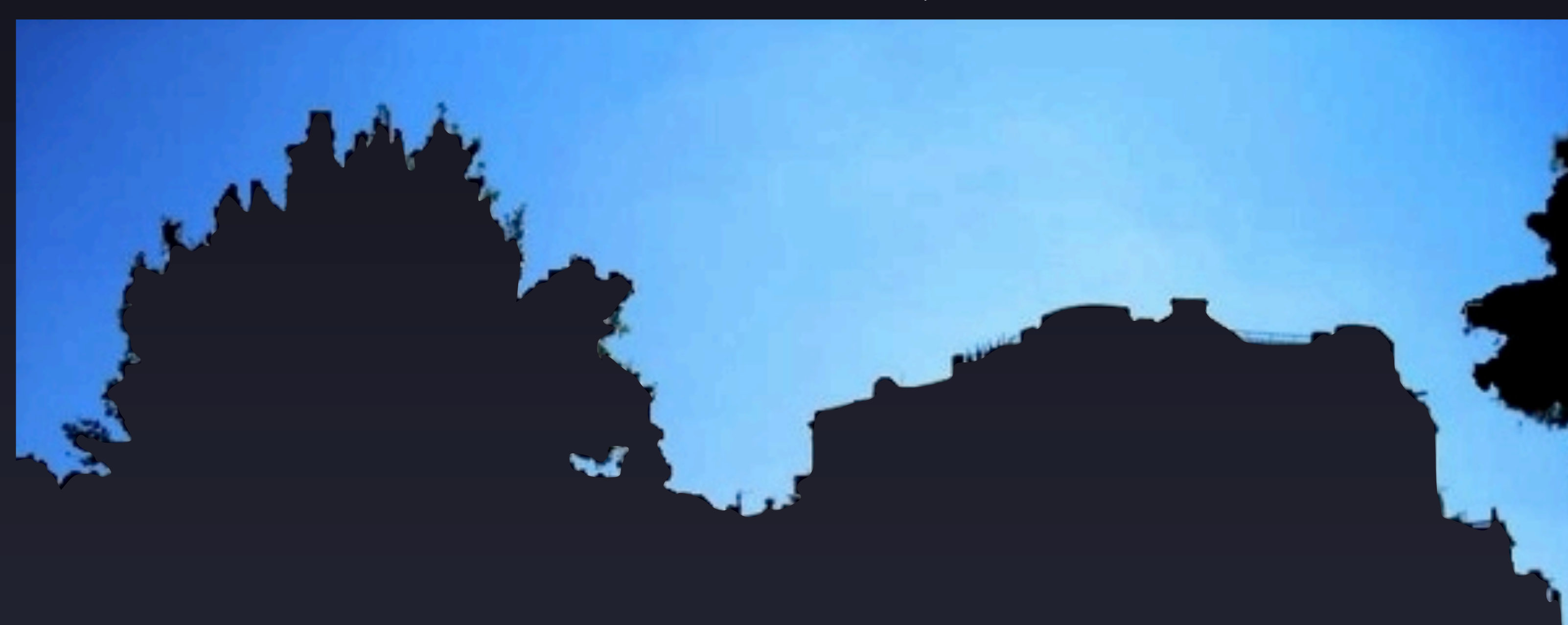


[Lalonde, Efros, and Narasimhan, ICCV 2009]

For a given sun position on the left, we synthesize the sky using the Perez sky model. We then model the probability of the sun having this particular position by

Sky

Original sky



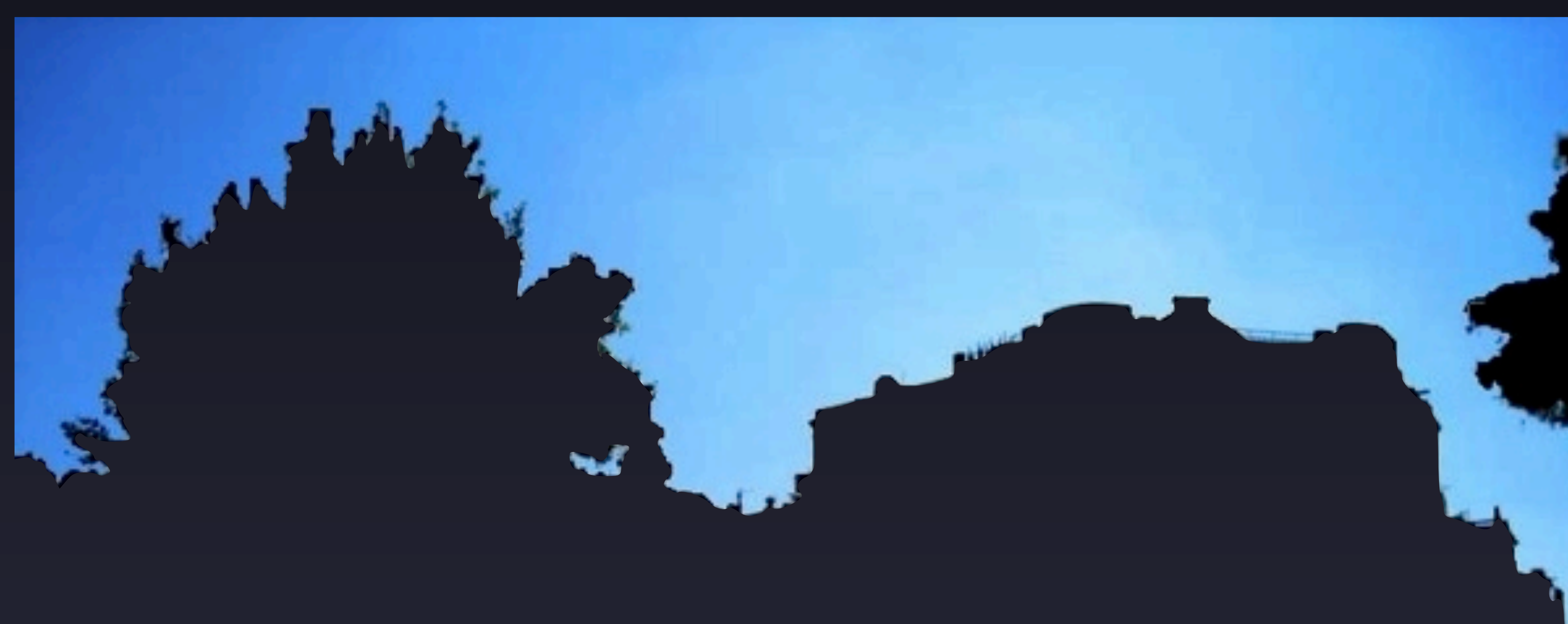
[Lalonde, Efros, and Narasimhan, ICCV 2009]

For a given sun position on the left, we synthesize the sky using the Perez sky model. We then model the probability of the sun having this particular position by

Sky

Predicted sky at current sun position

Original sky



[Lalonde, Efros, and Narasimhan, ICCV 2009]

For a given sun position on the left, we synthesize the sky using the Perez sky model. We then model the probability of the sun having this particular position by

Sky

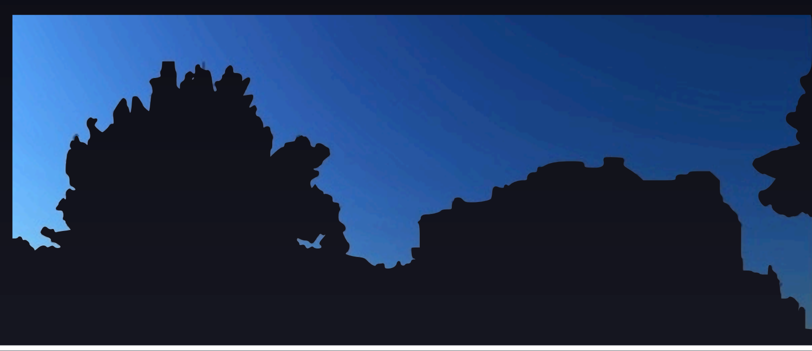
[Lalonde, Efros, and Narasimhan, ICCV 2009]

For a given sun position on the left, we synthesize the sky using the Perez sky model. We then model the probability of the sun having this particular position by

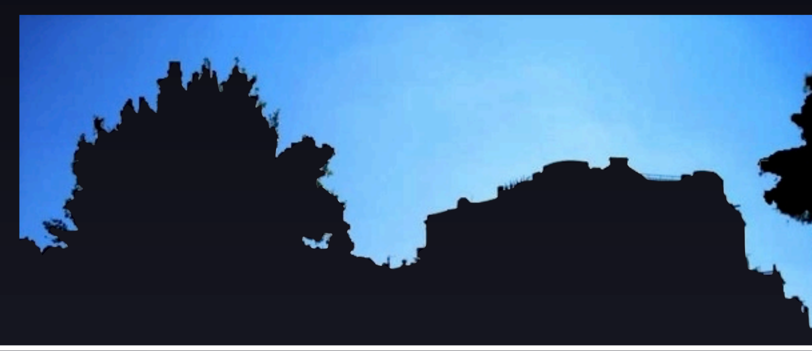
Sky probabilities

P(sun position | sky pixels)

exp $\dfrac{-\left(\quad\right.}{}$



\-



$\left.\quad\right)^2$

variance

[Lalonde, Efros, and Narasimhan, ICCV 2009]

summing the pixel-wise difference between the predicted and the actual sky, and taking the negative exponential. This way, we model a pixel with a gaussian centered at the value predicted by the model.

Sky

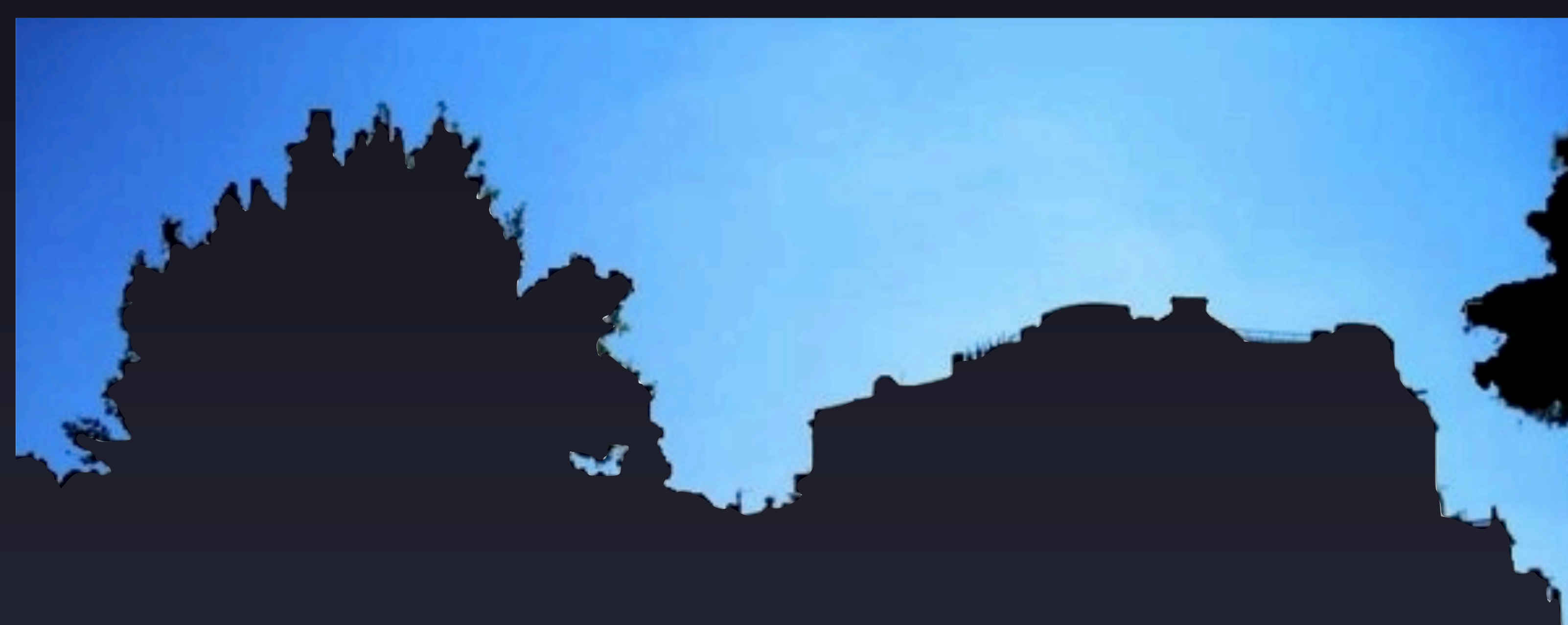
[Lalonde, Efros, and Narasimhan, ICCV 2009]

Let’s go back and try another sun position.
If we do this for every discretized sun position and normalize appropriately, we obtain our final probability map.

Sky

Predicted sky at current sun position

Original sky



[Lalonde, Efros, and Narasimhan, ICCV 2009]

Let’s go back and try another sun position.
If we do this for every discretized sun position and normalize appropriately, we obtain our final probability map.

Sky

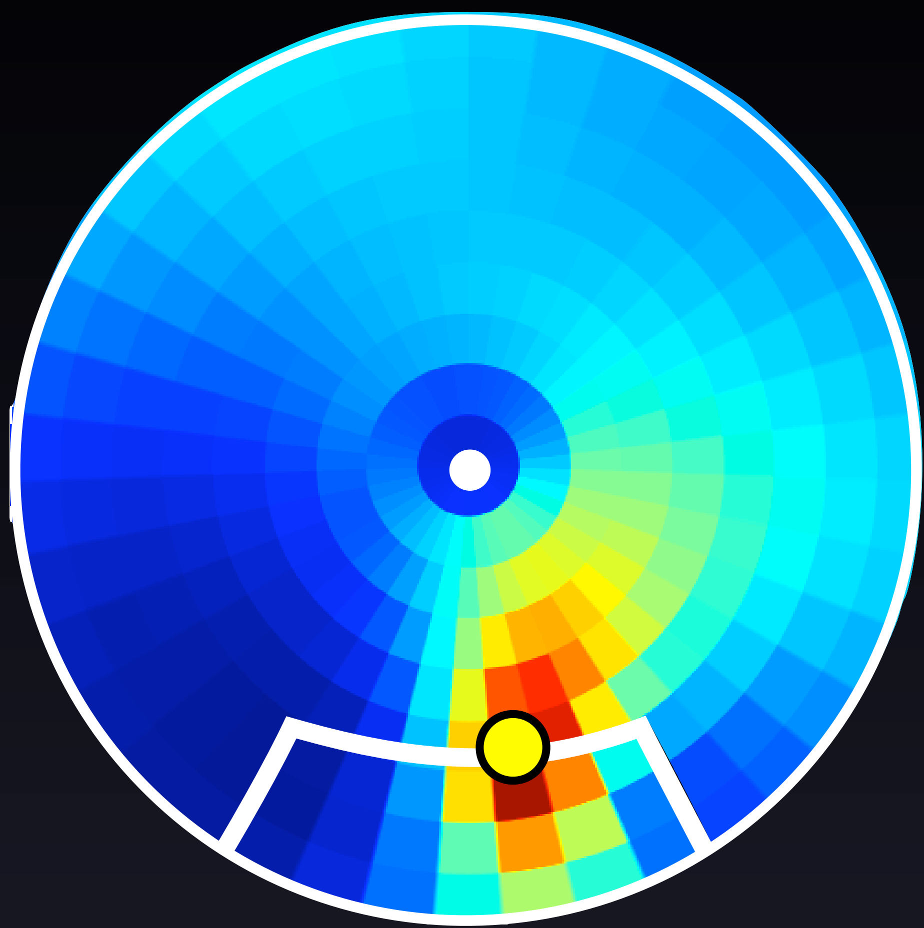
Predicted sky at current sun position

Original sky

[Lalonde, Efros, and Narasimhan, ICCV 2009]

Let’s go back and try another sun position.
If we do this for every discretized sun position and normalize appropriately, we obtain our final probability map.

Sky



[Lalonde, Efros, and Narasimhan, ICCV 2009]

Let’s go back and try another sun position.
If we do this for every discretized sun position and normalize appropriately, we obtain our final probability map.

What if the sky is not clear?

[Lalonde, Efros, and Narasimhan, ICCV 2009]

This works when the sky is clear, so what if there are clouds? or if it's overcast?

Clear vs overcast vs patchy

show image where clouds
are subtracted?

show the one on 23.

[Lalonde, Efros, and Narasimhan, ICCV 2009]

Compare the color histogram of the sky with a sky database of 3 classes: clear, overcast, and patchy clouds. We use a k-nearest-neighbor classifier to decide which class it belongs to.

If the overcast class wins, we don't do any fitting and declare the sky to be uninformative. If the sky is patchy, we first perform a simple color-based k-means segmentation with 2 clusters, and keep pixels which belong to the "bluest" cluster.

Clear vs overcast vs patchy

Clear

show image where clouds
are subtracted?

show the one on 23.

[Lalonde, Efros, and Narasimhan, ICCV 2009]

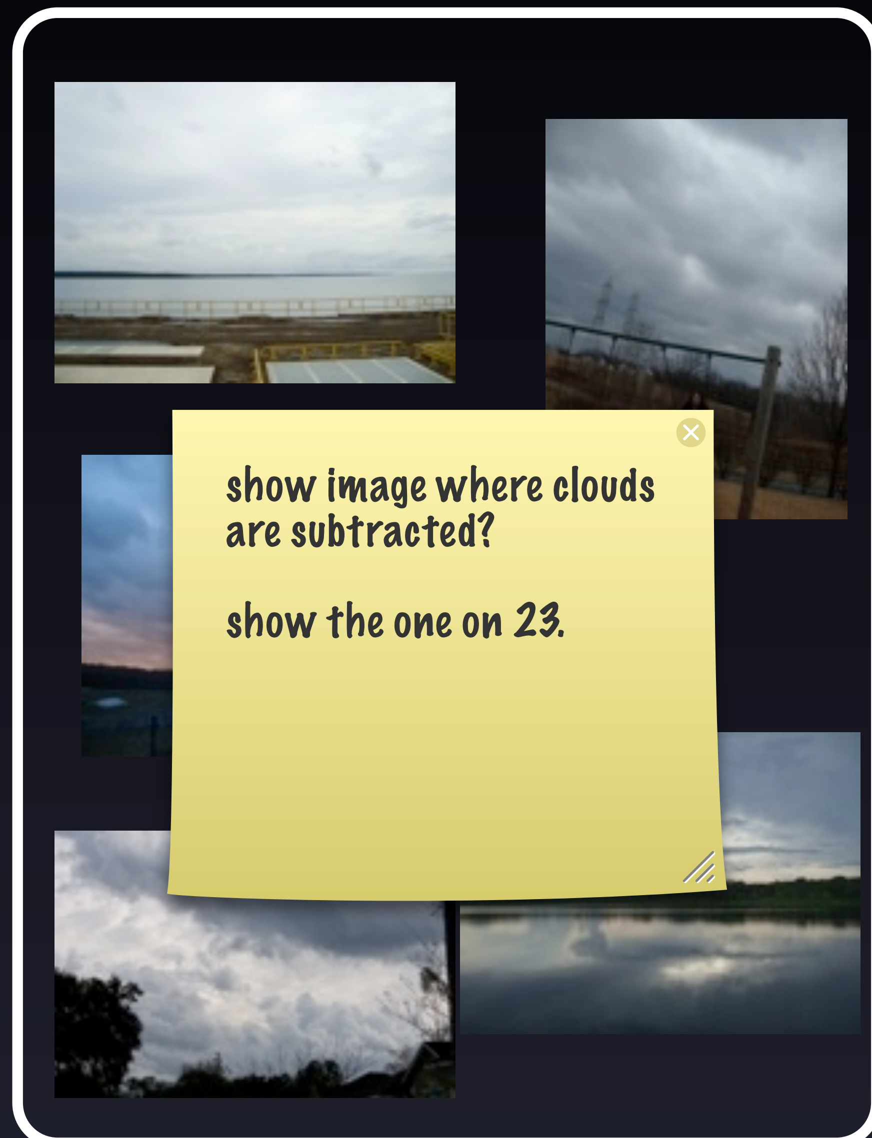
Compare the color histogram of the sky with a sky database of 3 classes: clear, overcast, and patchy clouds. We use a k-nearest-neighbor classifier to decide which class it belongs to.

If the overcast class wins, we don't do any fitting and declare the sky to be uninformative. If the sky is patchy, we first perform a simple color-based k-means segmentation with 2 clusters, and keep pixels which belong to the "bluest" cluster.

Clear vs overcast vs patchy

Clear

Overcast



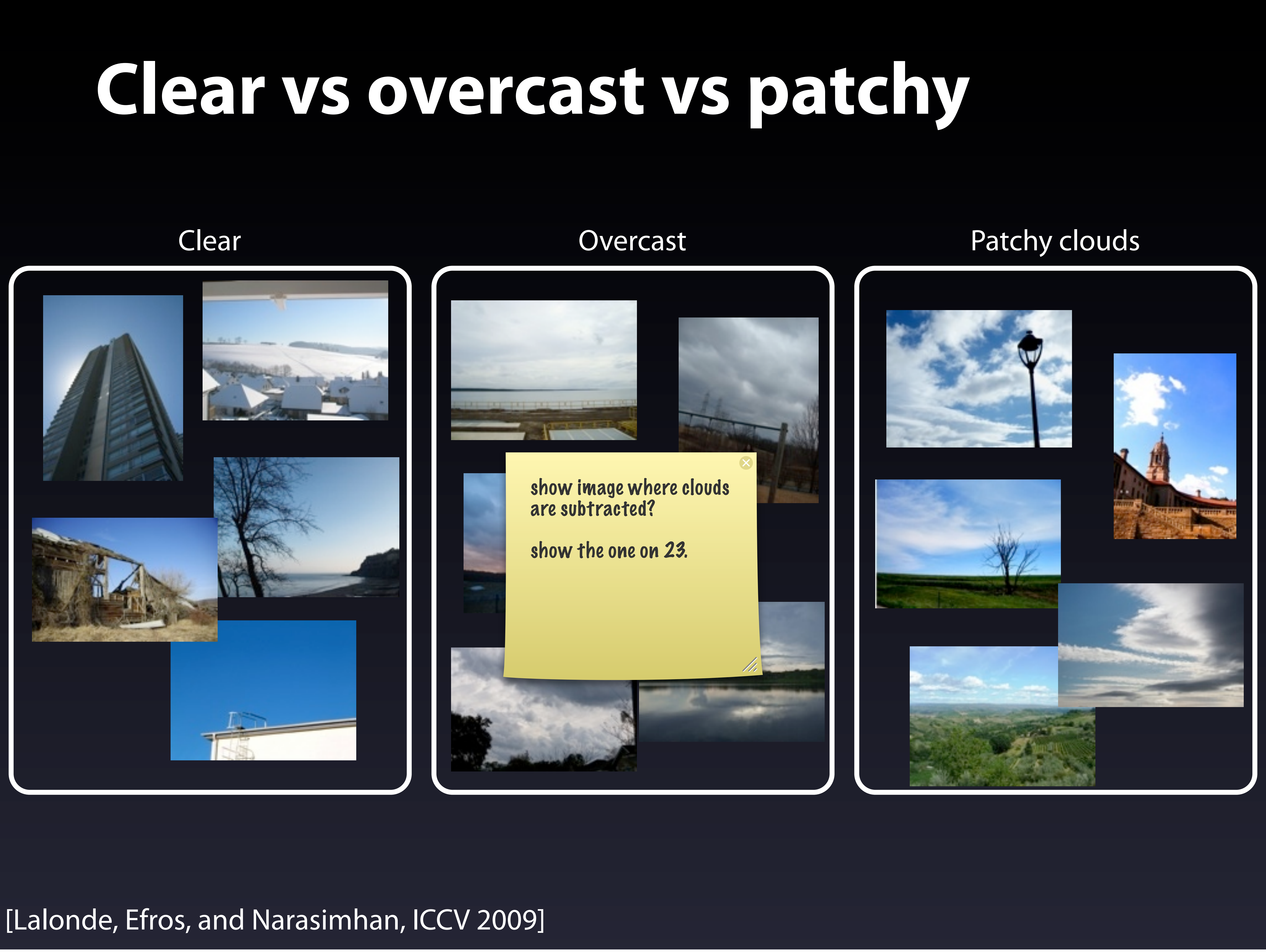
show image where clouds are subtracted?

show the one on 23.

[Lalonde, Efros, and Narasimhan, ICCV 2009]

Compare the color histogram of the sky with a sky database of 3 classes: clear, overcast, and patchy clouds. We use a k-nearest-neighbor classifier to decide which class it belongs to.

If the overcast class wins, we don't do any fitting and declare the sky to be uninformative. If the sky is patchy, we first perform a simple color-based k-means segmentation with 2 clusters, and keep pixels which belong to the "bluest" cluster.



Compare the color histogram of the sky with a sky database of 3 classes: clear, overcast, and patchy clouds. We use a k-nearest-neighbor classifier to decide which class it belongs to.

If the overcast class wins, we don't do any fitting and declare the sky to be uninformative. If the sky is patchy, we first perform a simple color-based k-means segmentation with 2 clusters, and keep pixels which belong to the "bluest" cluster.

Sun position given sky

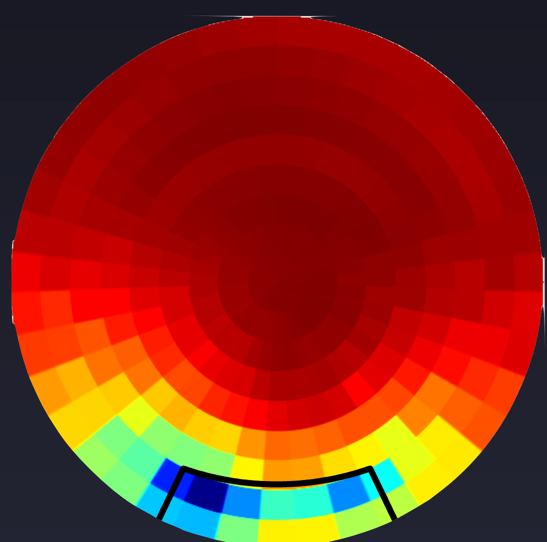
[Lalonde, Efros, and Narasimhan, ICCV 2009]

Sun position given sky

[Lalonde, Efros, and Narasimhan, ICCV 2009]

Limitations of the sky cue

Sun behind camera

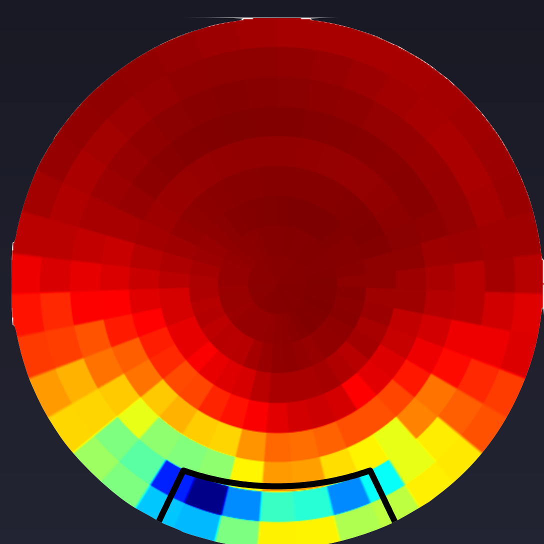


[Lalonde, Efros, and Narasimhan, ICCV 2009]

Unfortunately, the sky is not always helpful. For instance, when the sun is behind the camera, the sky is very uncertain about its position: it might be anywhere except in the camera field of view. Sometimes, too, the sky might not even be visible, in which case this cue returns a constant probability map. But notice that in these two images, the shadows seem to offer information about the sun position, couldn't we exploit those instead?

Limitations of the sky cue

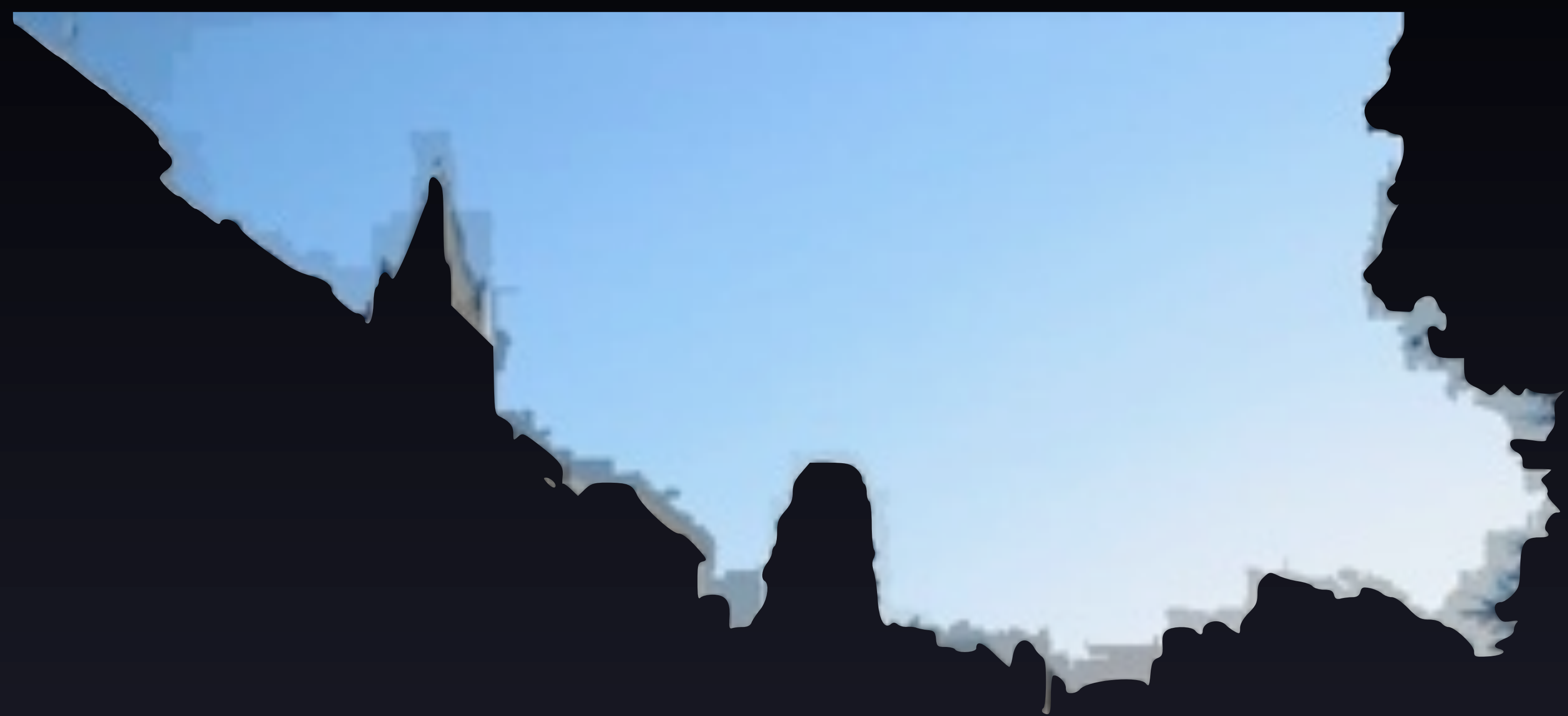
Sun behind camera



Sky not visible

[Lalonde, Efros, and Narasimhan, ICCV 2009]

Unfortunately, the sky is not always helpful. For instance, when the sun is behind the camera, the sky is very uncertain about its position: it might be anywhere except in the camera field of view. Sometimes, too, the sky might not even be visible, in which case this cue returns a constant probability map. But notice that in these two images, the shadows seem to offer information about the sun position, couldn't we exploit those instead?



[Lalonde, Efros, and Narasimhan, ICCV 2009]

Now that we've seen how we can exploit the sky, let's consider another important illumination cue: shadows cast on the ground.

Sky

Ground

Vertical surfaces

[Lalonde, Efros, and Narasimhan, ICCV 2009]

Now that we've seen how we can exploit the sky, let's consider another important illumination cue: shadows cast on the ground.

[Lalonde, Efros, and Narasimhan, ICCV 2009]

Now that we’ve seen how we can exploit the sky, let’s consider another important illumination cue: shadows cast on the ground.

Cast shadows

[Lalonde, Efros, and Narasimhan, ICCV 2009]

Vertical objects, such as this lamppost, act as sun dials. We know that their shadows, when cast on the ground, point towards the sun. However, doing this the right way implies that we would need to automatically detect the light post and its shadow, reason about its contact point with the ground, figure out its height, etc. Of course, this is extremely hard and nobody really knows how to do it. Instead we'll adopt a different approach and consider the statistical distribution of shadow lines on the ground.

We'll assume that on average, most of the cast shadows on the ground come from vertical objects, and that's reasonable since gravity makes a lot of objects stand up straight. Since we don't know the height of objects, we will predict sun azimuth only, and not the elevation. And we'll keep the directional ambiguity since we don't know the contact points with the ground.

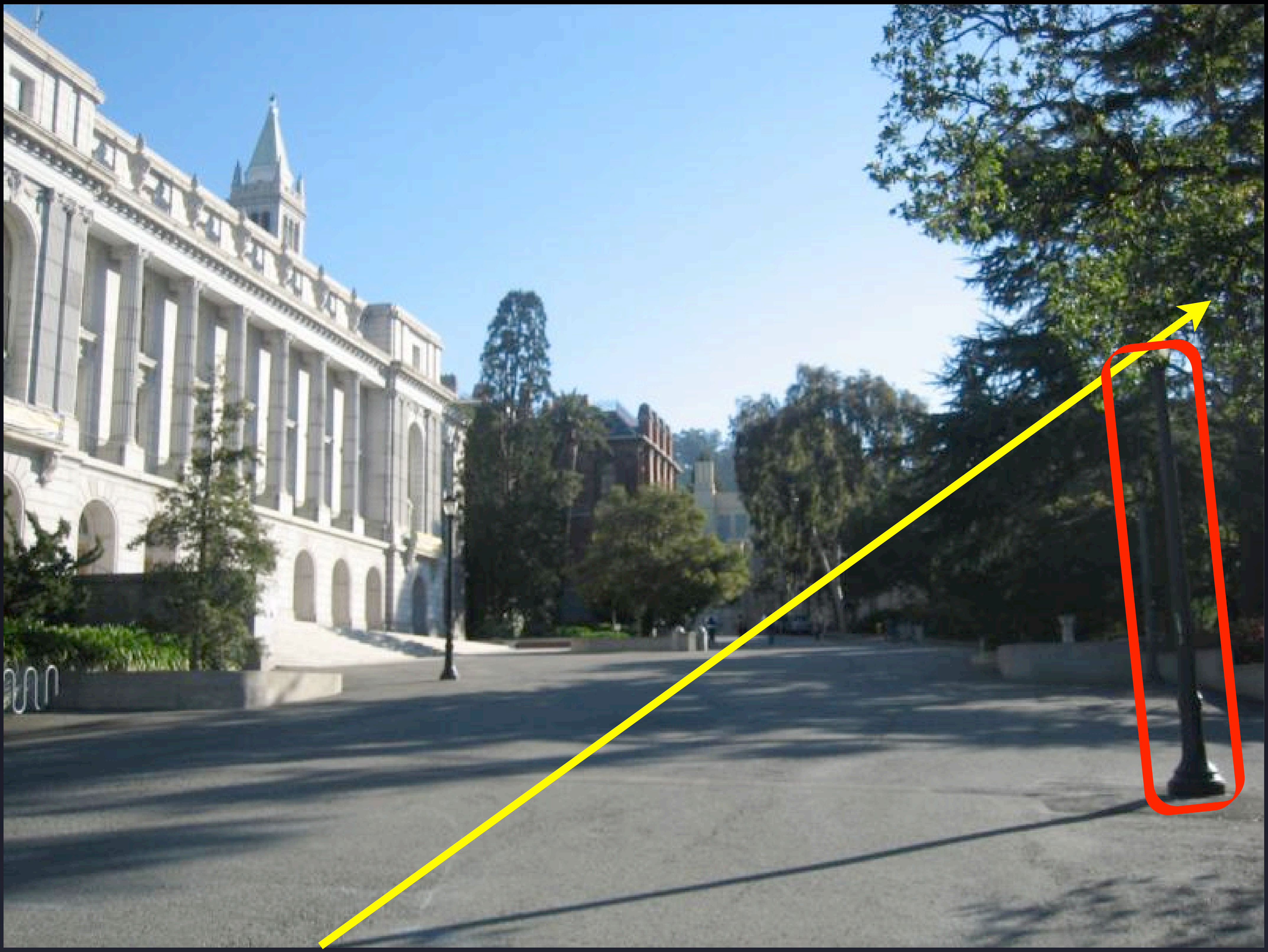
Cast shadows

[Lalonde, Efros, and Narasimhan, ICCV 2009]

Vertical objects, such as this lamppost, act as sun dials. We know that their shadows, when cast on the ground, point towards the sun. However, doing this the right way implies that we would need to automatically detect the light post and its shadow, reason about its contact point with the ground, figure out its height, etc. Of course, this is extremely hard and nobody really knows how to do it. Instead we'll adopt a different approach and consider the statistical distribution of shadow lines on the ground.

We'll assume that on average, most of the cast shadows on the ground come from vertical objects, and that's reasonable since gravity makes a lot of objects stand up straight. Since we don't know the height of objects, we will predict sun azimuth only, and not the elevation. And we'll keep the directional ambiguity since we don't know the contact points with the ground.

Cast shadows



[Lalonde, Efros, and Narasimhan, ICCV 2009]

Vertical objects, such as this lamppost, act as sun dials. We know that their shadows, when cast on the ground, point towards the sun. However, doing this the right way implies that we would need to automatically detect the light post and its shadow, reason about its contact point with the ground, figure out its height, etc. Of course, this is extremely hard and nobody really knows how to do it. Instead we'll adopt a different approach and consider the statistical distribution of shadow lines on the ground.

We'll assume that on average, most of the cast shadows on the ground come from vertical objects, and that's reasonable since gravity makes a lot of objects stand up straight. Since we don't know the height of objects, we will predict sun azimuth only, and not the elevation. And we'll keep the directional ambiguity since we don't know the contact points with the ground.

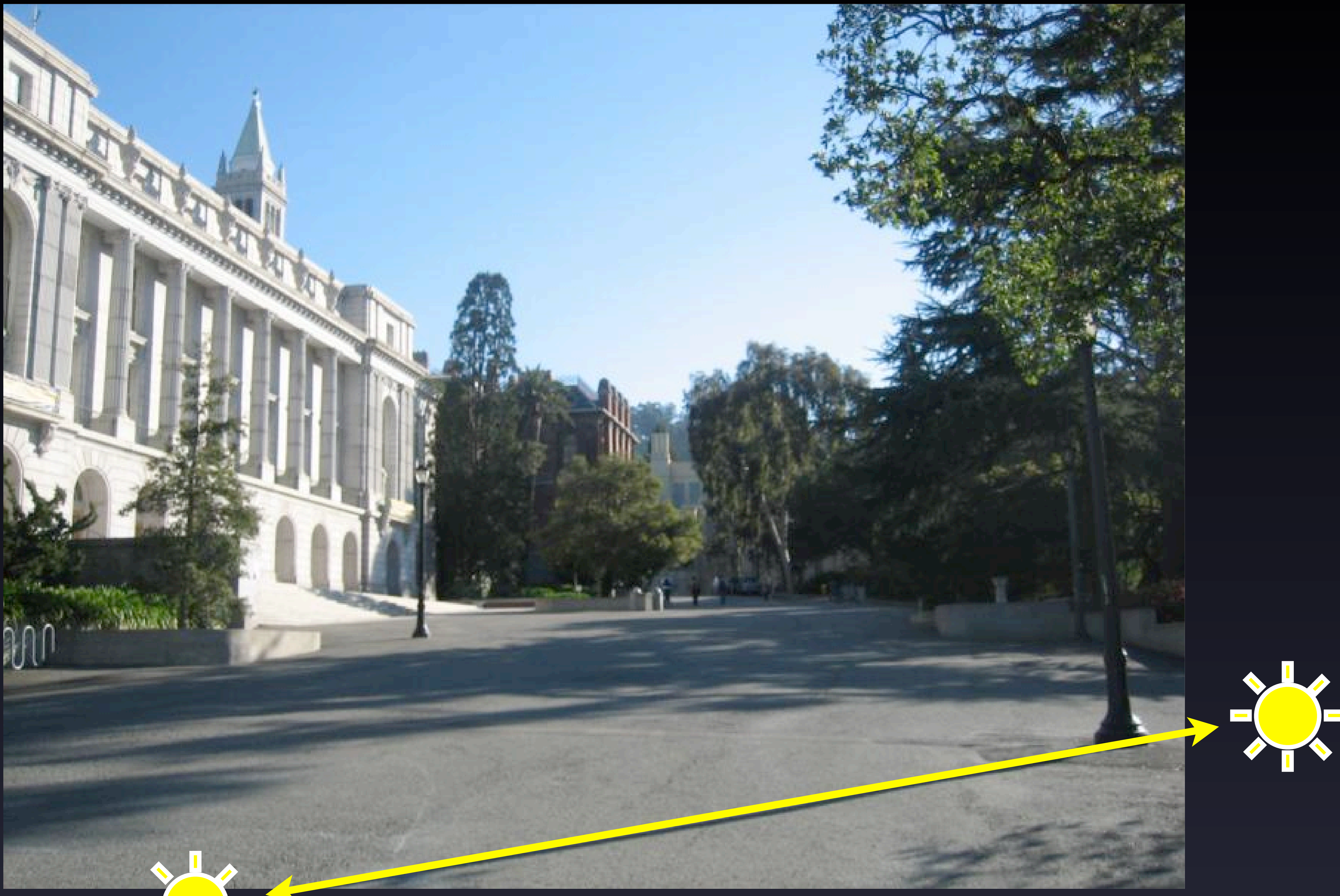
Cast shadows

[Lalonde, Efros, and Narasimhan, ICCV 2009]

Vertical objects, such as this lamppost, act as sun dials. We know that their shadows, when cast on the ground, point towards the sun. However, doing this the right way implies that we would need to automatically detect the light post and its shadow, reason about its contact point with the ground, figure out its height, etc. Of course, this is extremely hard and nobody really knows how to do it. Instead we'll adopt a different approach and consider the statistical distribution of shadow lines on the ground.

We'll assume that on average, most of the cast shadows on the ground come from vertical objects, and that's reasonable since gravity makes a lot of objects stand up straight. Since we don't know the height of objects, we will predict sun azimuth only, and not the elevation. And we'll keep the directional ambiguity since we don't know the contact points with the ground.

Cast shadows



[Lalonde, Efros, and Narasimhan, ICCV 2009]

Vertical objects, such as this lamppost, act as sun dials. We know that their shadows, when cast on the ground, point towards the sun. However, doing this the right way implies that we would need to automatically detect the light post and its shadow, reason about its contact point with the ground, figure out its height, etc. Of course, this is extremely hard and nobody really knows how to do it. Instead we'll adopt a different approach and consider the statistical distribution of shadow lines on the ground.

We'll assume that on average, most of the cast shadows on the ground come from vertical objects, and that's reasonable since gravity makes a lot of objects stand up straight. Since we don't know the height of objects, we will predict sun azimuth only, and not the elevation. And we'll keep the directional ambiguity since we don't know the contact points with the ground.

Shadow detection

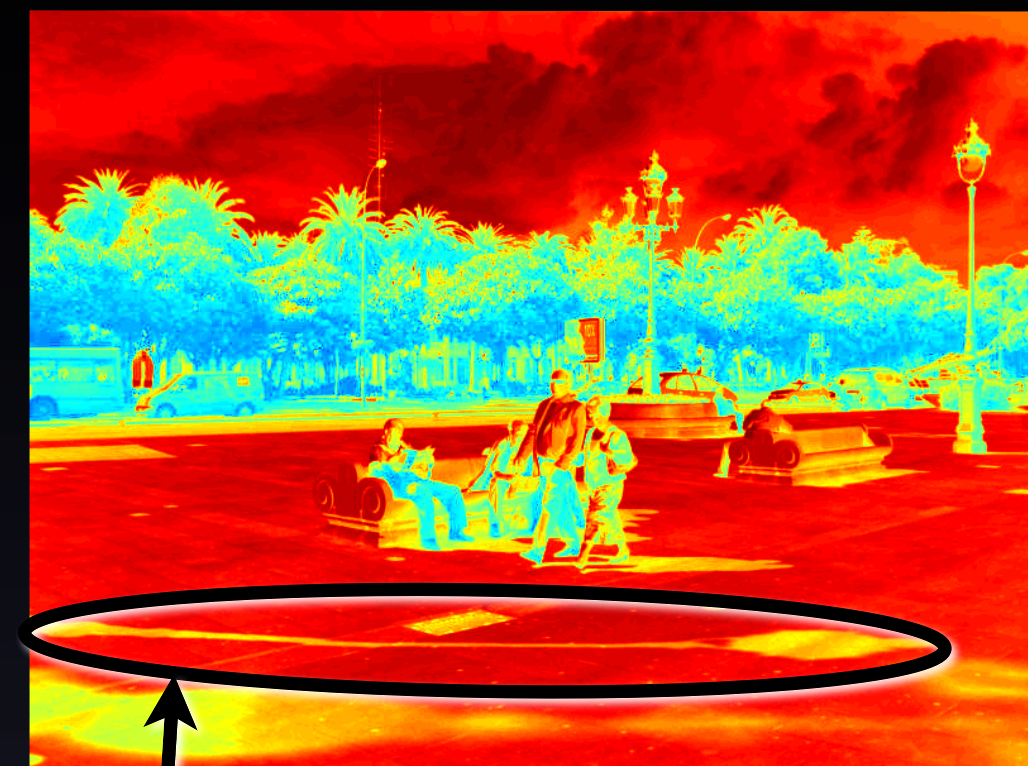
[Lalonde, Efros, and Narasimhan, ICCV 2009]

[Khan & Reinhard, ICIP '05]

We propose a series of simple steps which try to detect as many shadow edges as possible and keep the number of detected reflectance edges to a minimum.
First, (Khan and Reinhard) made the observation strong shadow edges are typically visible in the L channel, but not in the a channel, as opposed to reflectance edges which are visible in both. Based on that insight, we compute edges that are present in the L channel, but not in the a channel.
We then find long lines using the Video Compass approach, which should take care of filtering out short, noisy edges.

Shadow detection

L channel

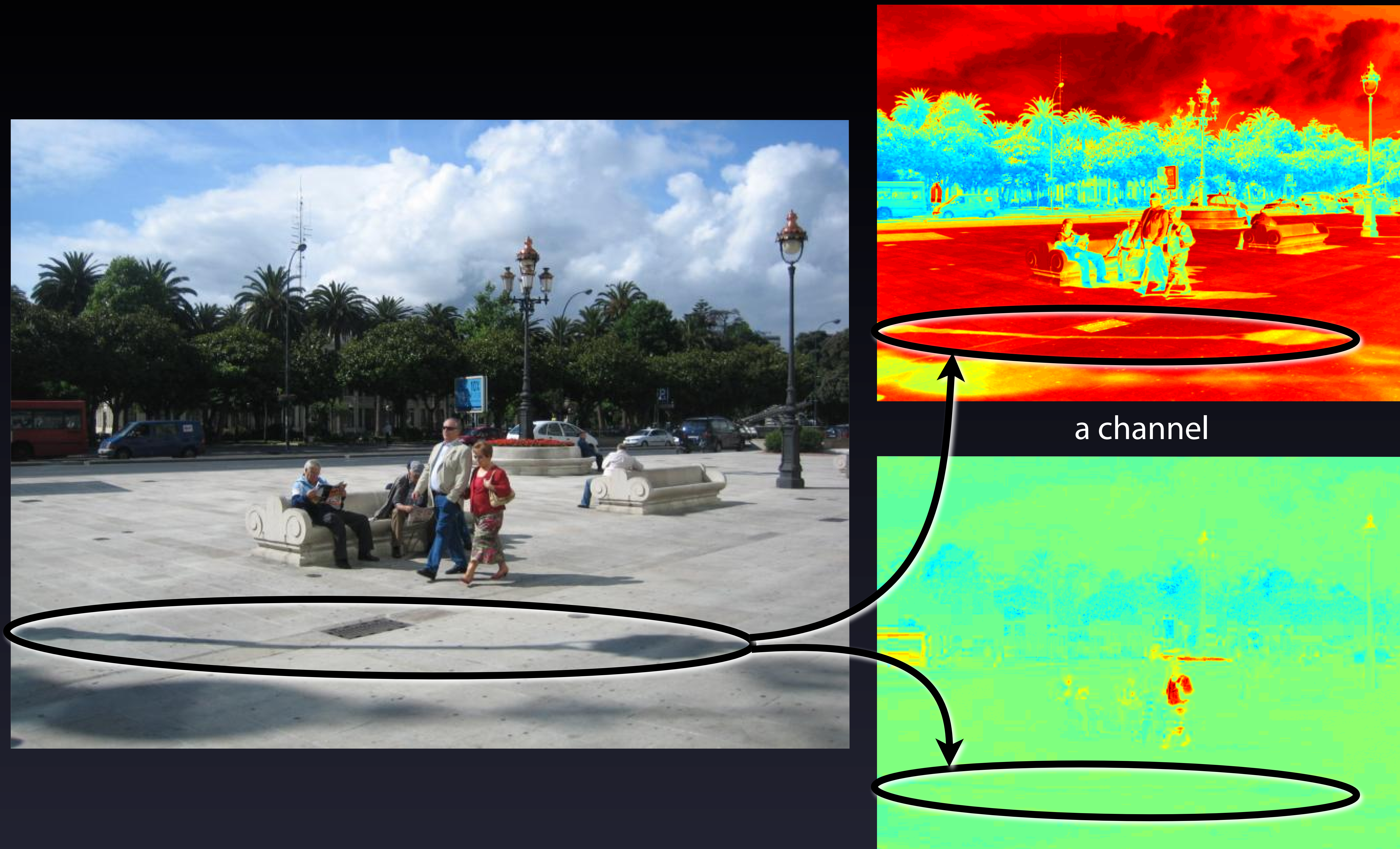


[Khan & Reinhard, ICIP '05]

[Lalonde, Efros, and Narasimhan, ICCV 2009]

We propose a series of simple steps which try to detect as many shadow edges as possible and keep the number of detected reflectance edges to a minimum.
First, (Khan and Reinhard) made the observation strong shadow edges are typically visible in the L channel, but not in the a channel, as opposed to reflectance edges which are visible in both. Based on that insight, we compute edges that are present in the L channel, but not in the a channel.
We then find long lines using the Video Compass approach, which should take care of filtering out short, noisy edges.

Shadow detection

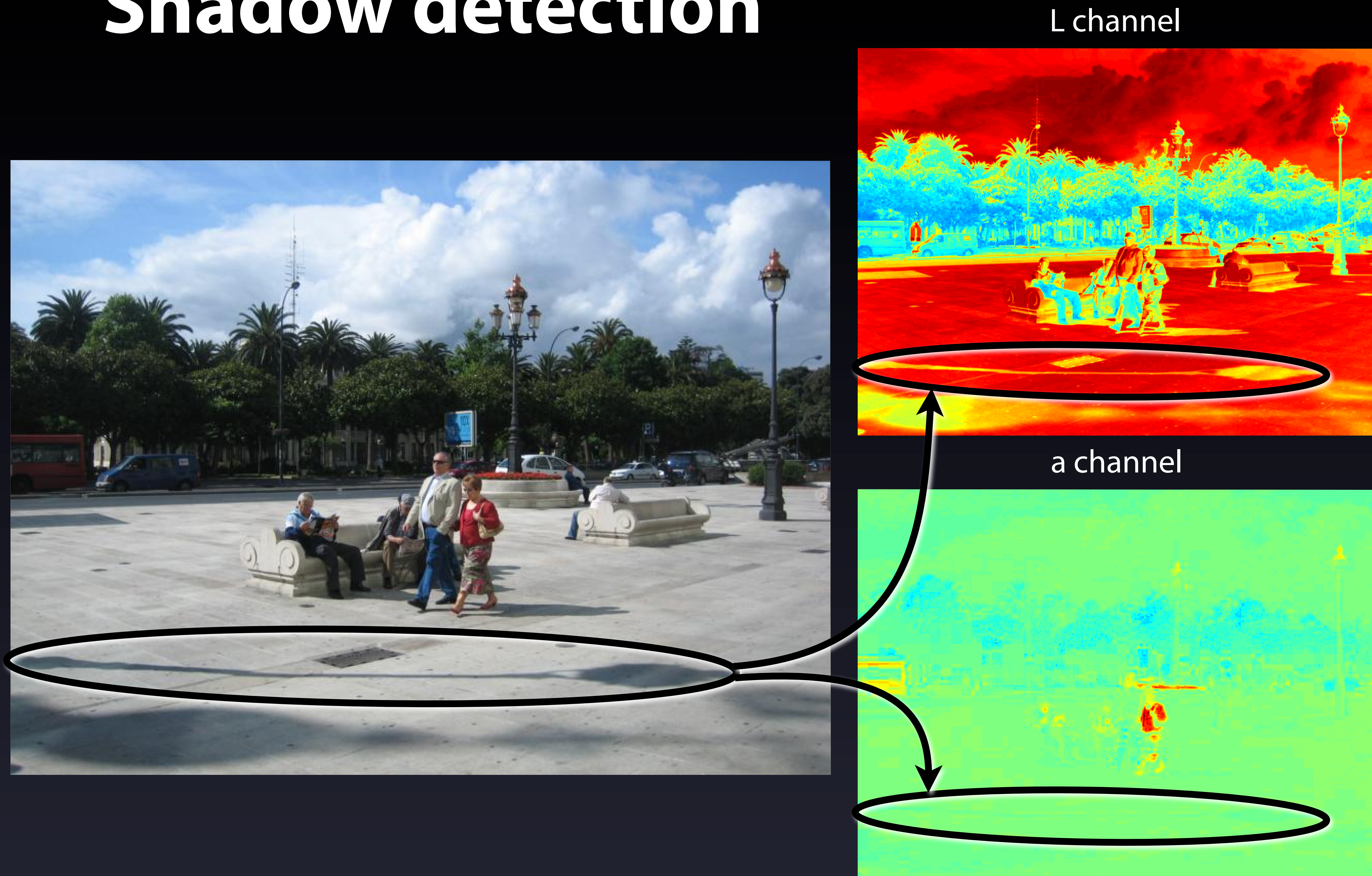


[Khan & Reinhard, ICIP ’05]

[Lalonde, Efros, and Narasimhan, ICCV 2009]

We propose a series of simple steps which try to detect as many shadow edges as possible and keep the number of detected reflectance edges to a minimum.
First, (Khan and Reinhard) made the observation strong shadow edges are typically visible in the L channel, but not in the a channel, as opposed to reflectance edges which are visible in both. Based on that insight, we compute edges that are present in the L channel, but not in the a channel.
We then find long lines using the Video Compass approach, which should take care of filtering out short, noisy edges.

Shadow detection



[Lalonde, Efros, and Narasimhan, ICCV 2009]

[Khan & Reinhard, ICIP ’05]
[Kosecka & Wang, ECCV ’02]

We propose a series of simple steps which try to detect as many shadow edges as possible and keep the number of detected reflectance edges to a minimum.
First, (Khan and Reinhard) made the observation strong shadow edges are typically visible in the L channel, but not in the a channel, as opposed to reflectance edges which are visible in both. Based on that insight, we compute edges that are present in the L channel, but not in the a channel.
We then find long lines using the Video Compass approach, which should take care of filtering out short, noisy edges.

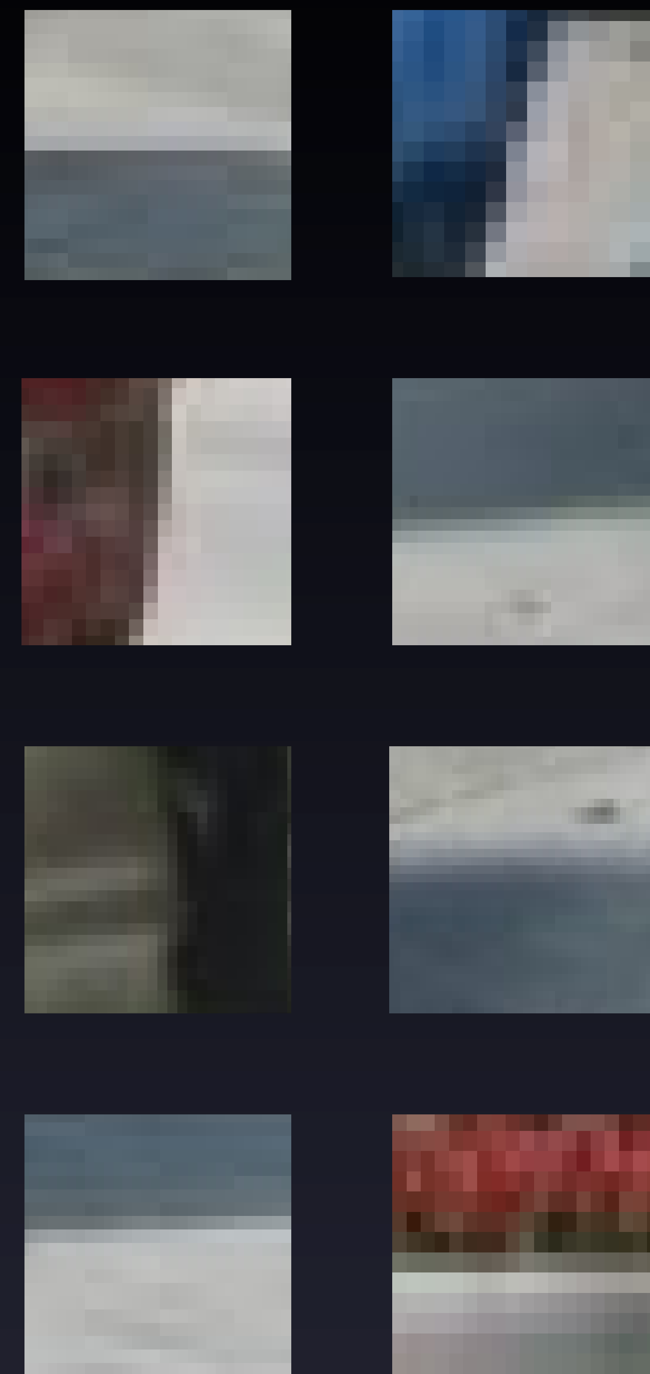
Shadow detection

[Lalonde, Efros, and Narasimhan, ICCV 2009]

Here's an example of some of the extracted edges using this technique. Looking at them more closely, we realize that shadow edges look very similar across the image, which is not the case of reflectance edges.

Shadow detection

Extracted edges

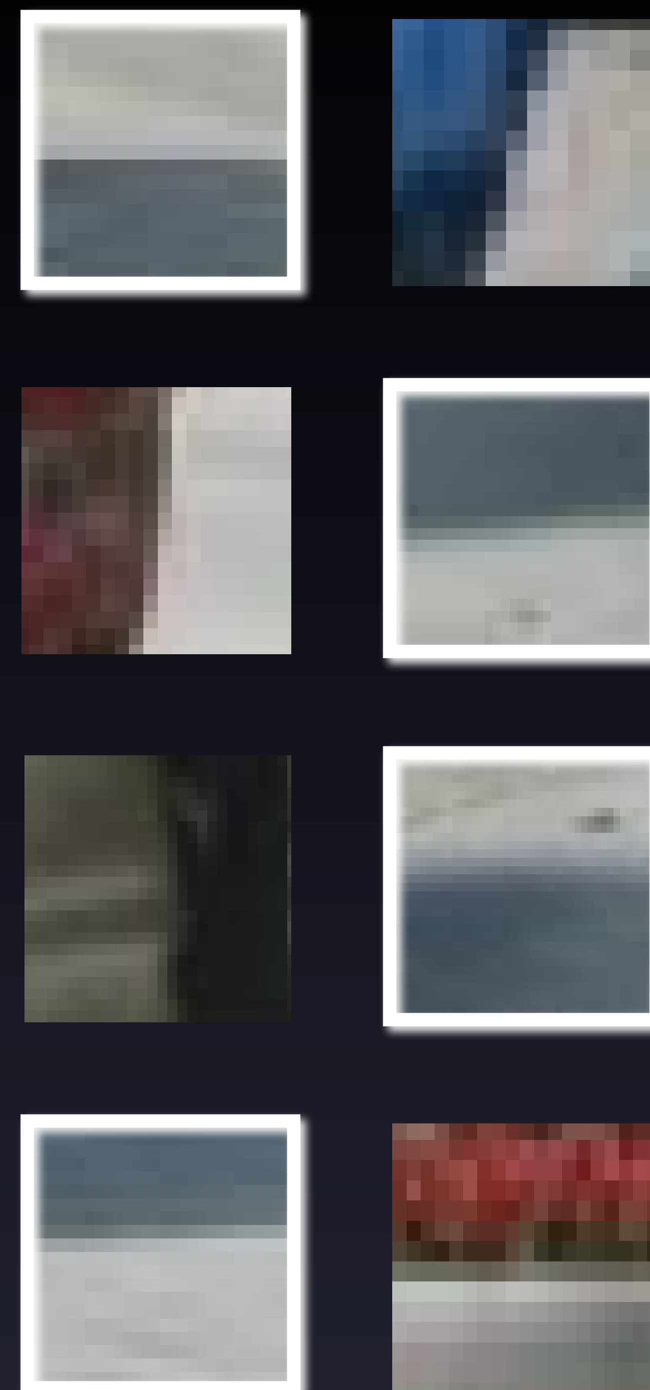


[Lalonde, Efros, and Narasimhan, ICCV 2009]

Here's an example of some of the extracted edges using this technique. Looking at them more closely, we realize that shadow edges look very similar across the image, which is not the case of reflectance edges.

Shadow detection

Extracted edges

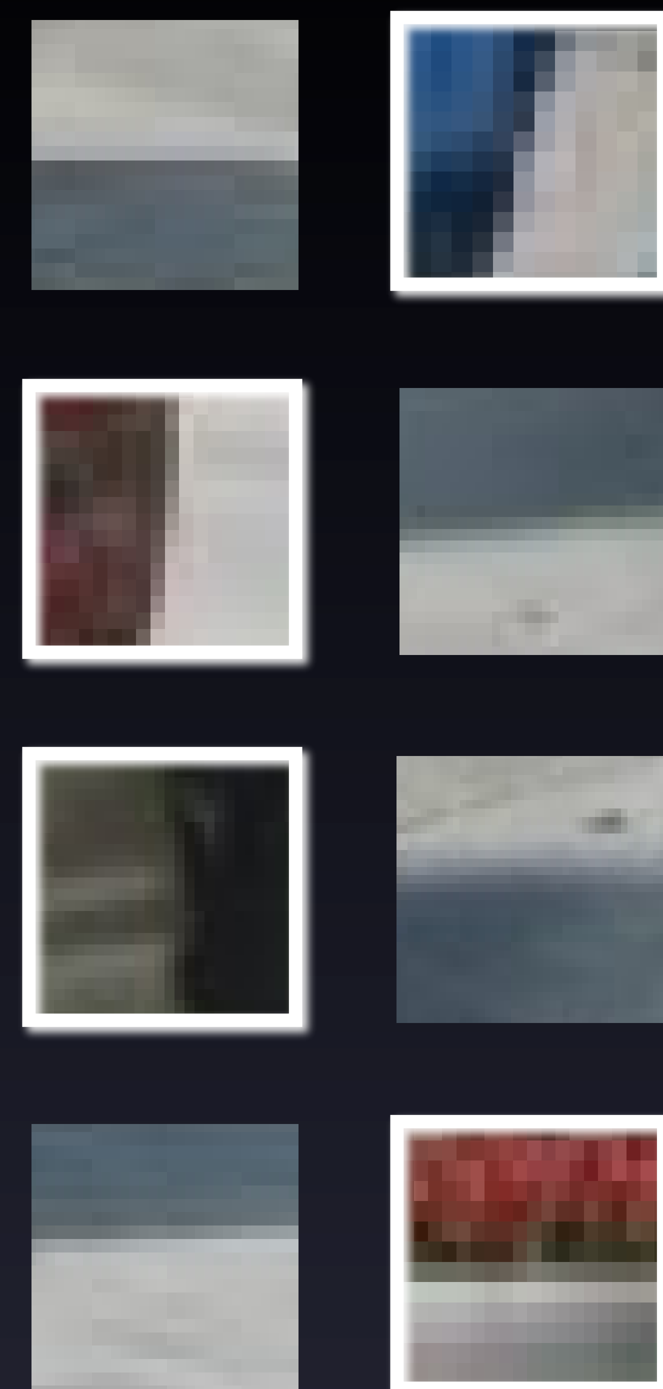


[Lalonde, Efros, and Narasimhan, ICCV 2009]

Here's an example of some of the extracted edges using this technique. Looking at them more closely, we realize that shadow edges look very similar across the image, which is not the case of reflectance edges.

Shadow detection

Extracted edges

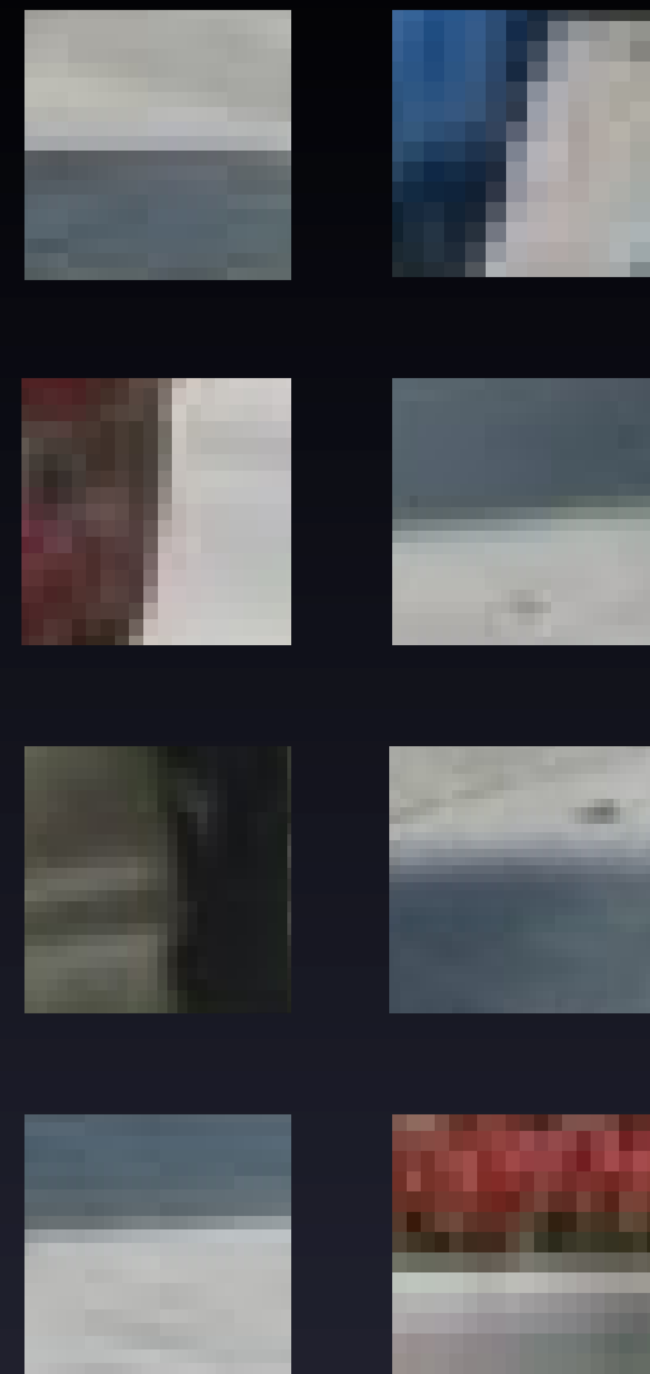


[Lalonde, Efros, and Narasimhan, ICCV 2009]

Here's an example of some of the extracted edges using this technique. Looking at them more closely, we realize that shadow edges look very similar across the image, which is not the case of reflectance edges.

Shadow detection

Extracted edges



[Lalonde, Efros, and Narasimhan, ICCV 2009]

By clustering them based on their appearance, we can group all the shadow edges together, and throw the small clusters away.

Shadow detection

[Lalonde, Efros, and Narasimhan, ICCV 2009]

By clustering them based on their appearance, we can group all the shadow edges together, and throw the small clusters away.

Shadow detection

[Lalonde, Efros, and Narasimhan, ICCV 2009]

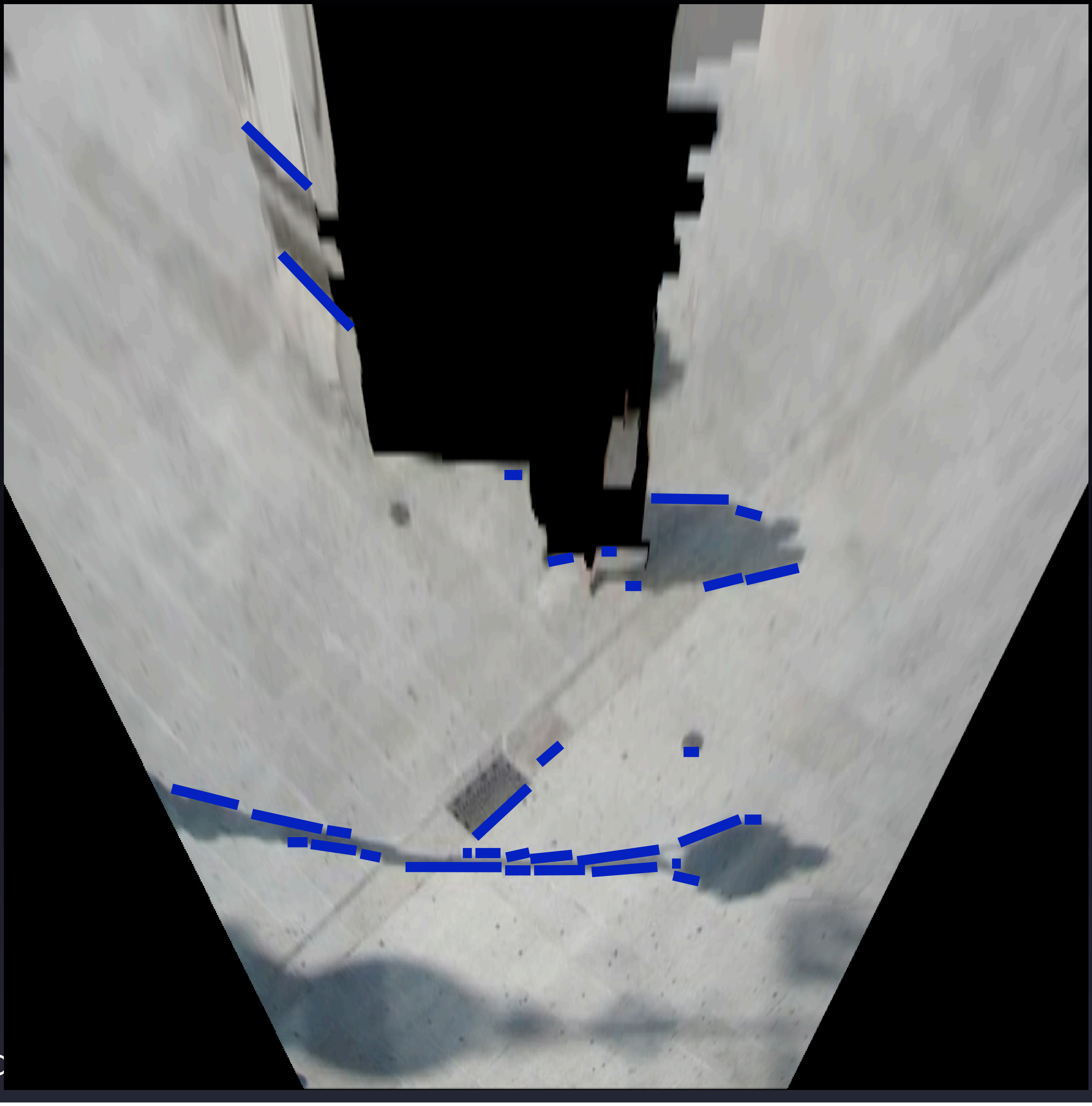
By clustering them based on their appearance, we can group all the shadow edges together, and throw the small clusters away.

Shadows

[Lalonde

And here is the final detection result. Note that we do not detect all shadow edges, but what's important is that we do not have many false positives. But how do we go from detected shadow lines to sun probability? First, we warp them in a top-down view using the focal length commonly available in EXIF tags. If we focus on a single shadow edge for now,

Shadows



[Lalonde, Efros, and

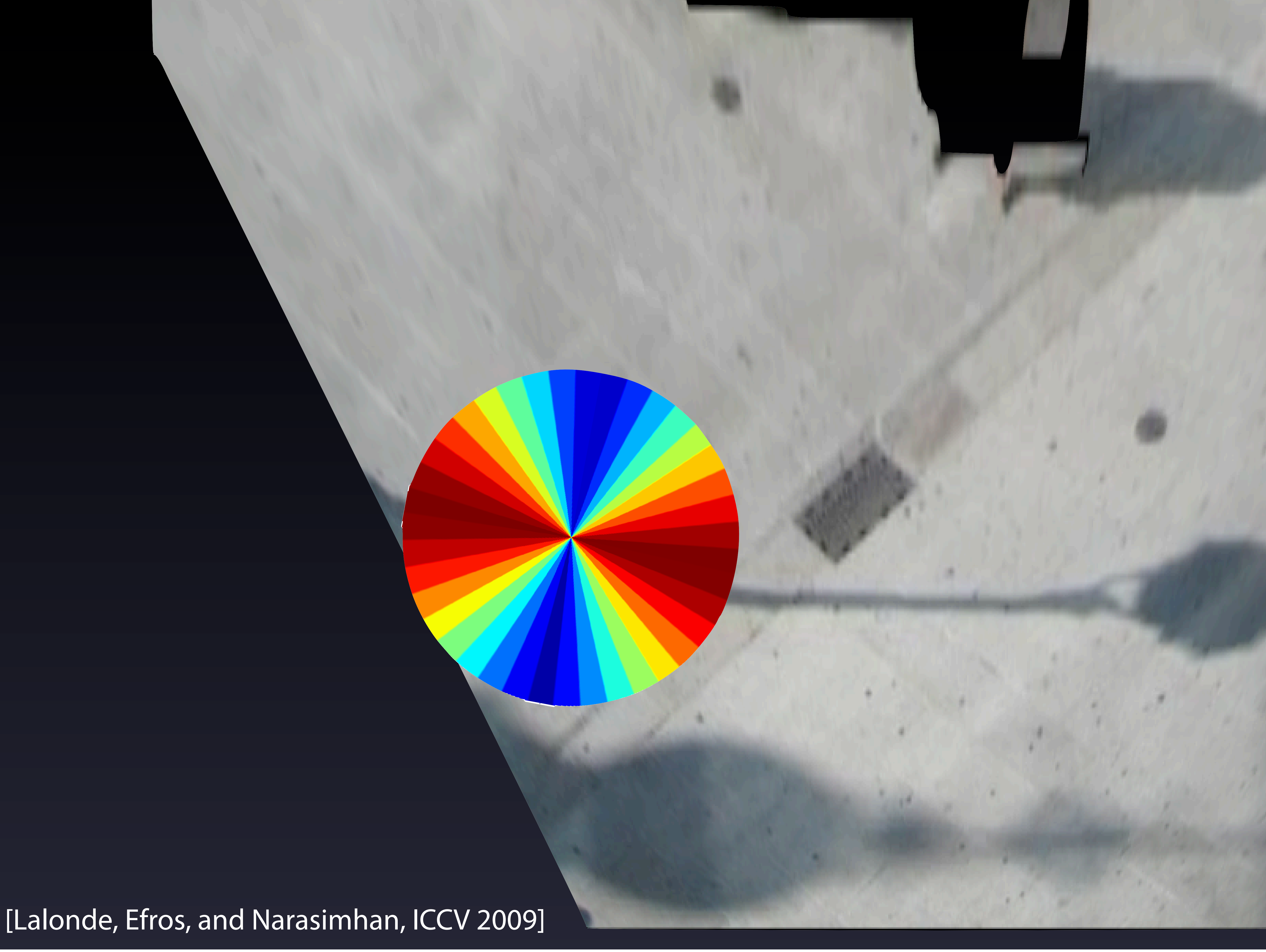
And here is the final detection result. Note that we do not detect all shadow edges, but what's important is that we do not have many false positives. But how do we go from detected shadow lines to sun probability? First, we warp them in a top-down view using the focal length commonly available in EXIF tags. If we focus on a single shadow edge for now,

[Lalonde, Efros, and Narasimhan, ICC

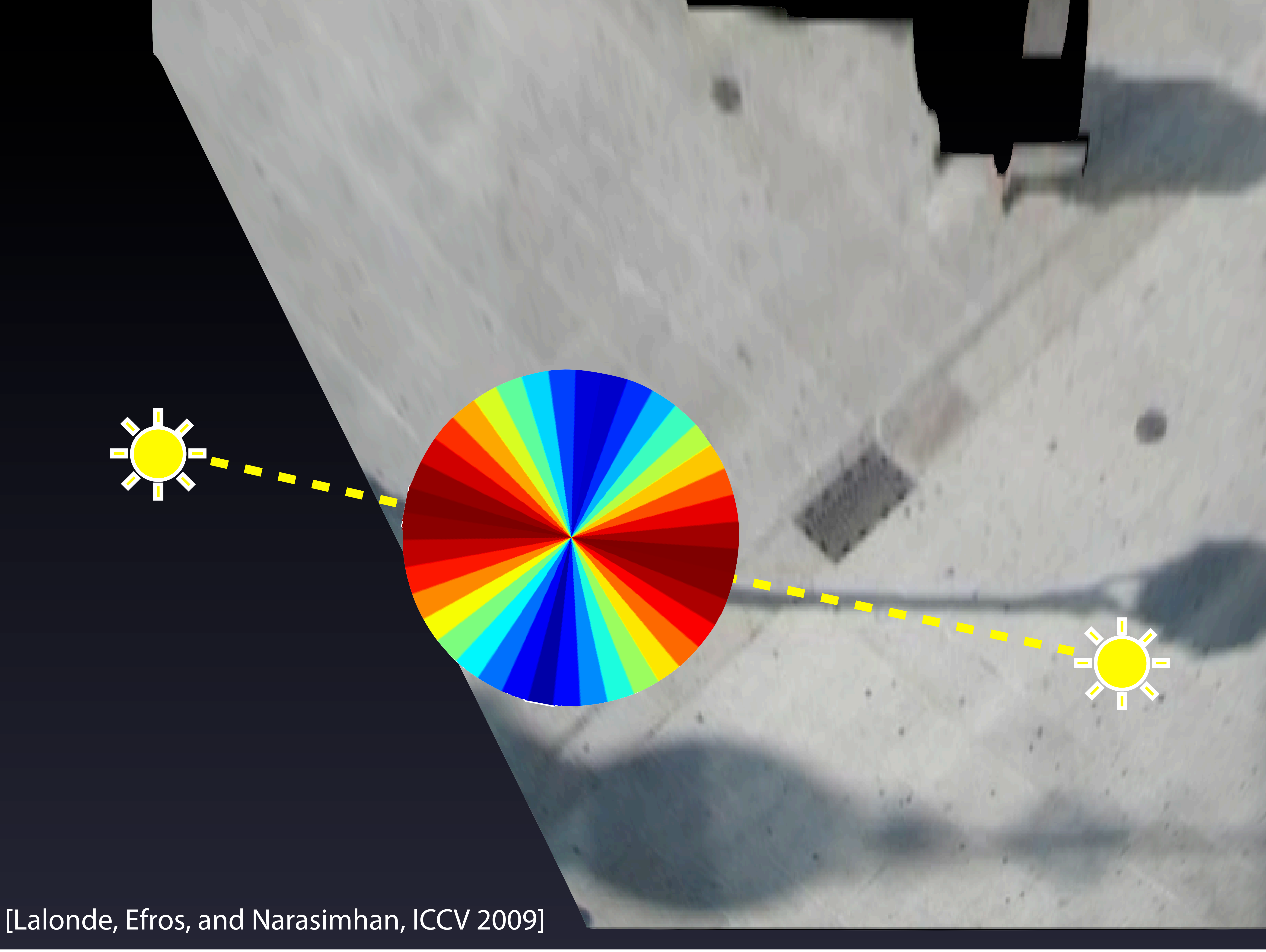
we model the sun azimuth distribution with 2 gaussians, separated by 180 degrees, since we have the directional ambiguity: the sun could either be here or here.

[Lalonde, Efros, and Narasimhan, ICCV 2009]

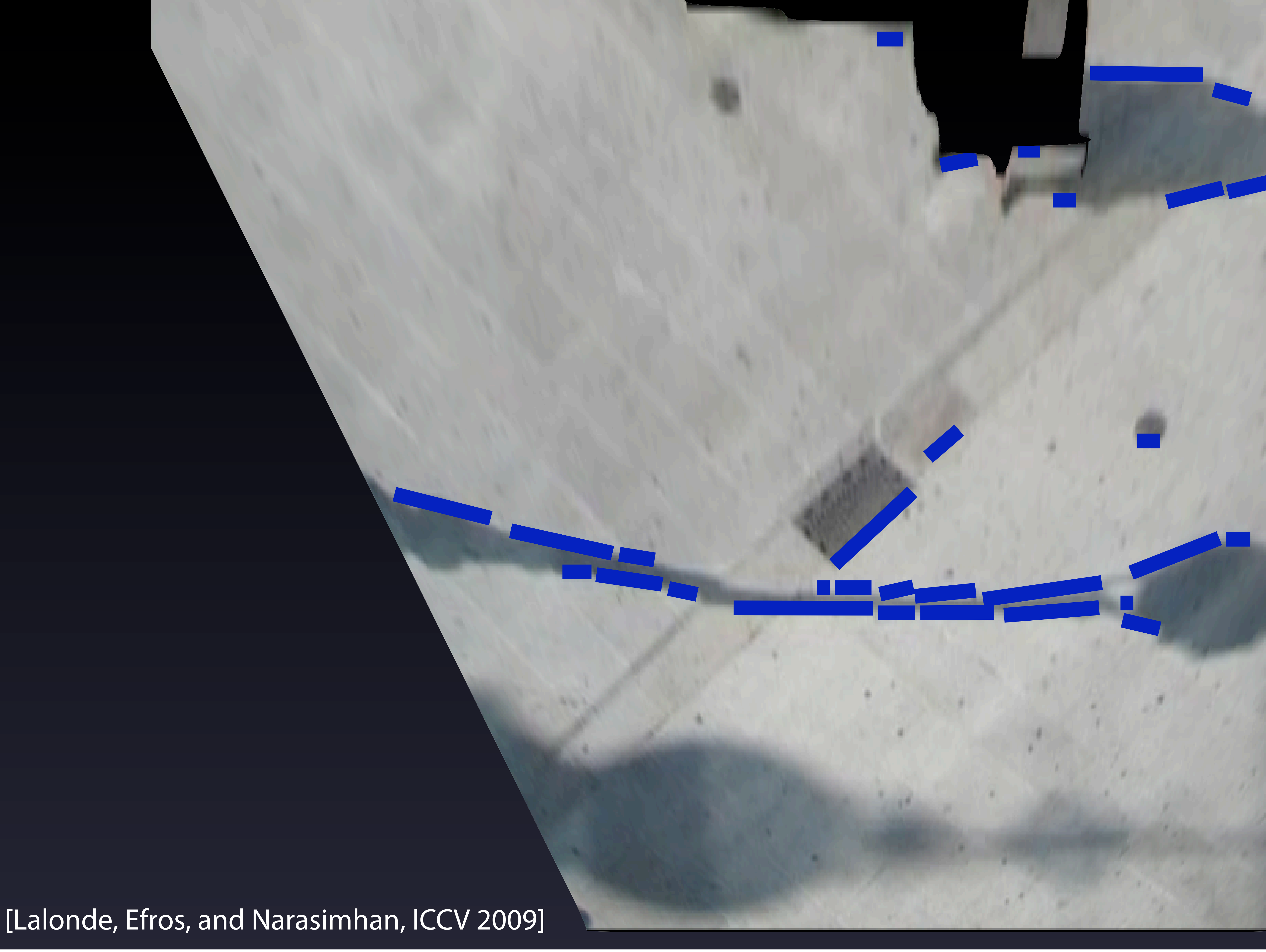
we model the sun azimuth distribution with 2 gaussians, separated by 180 degrees, since we have the directional ambiguity: the sun could either be here or here.



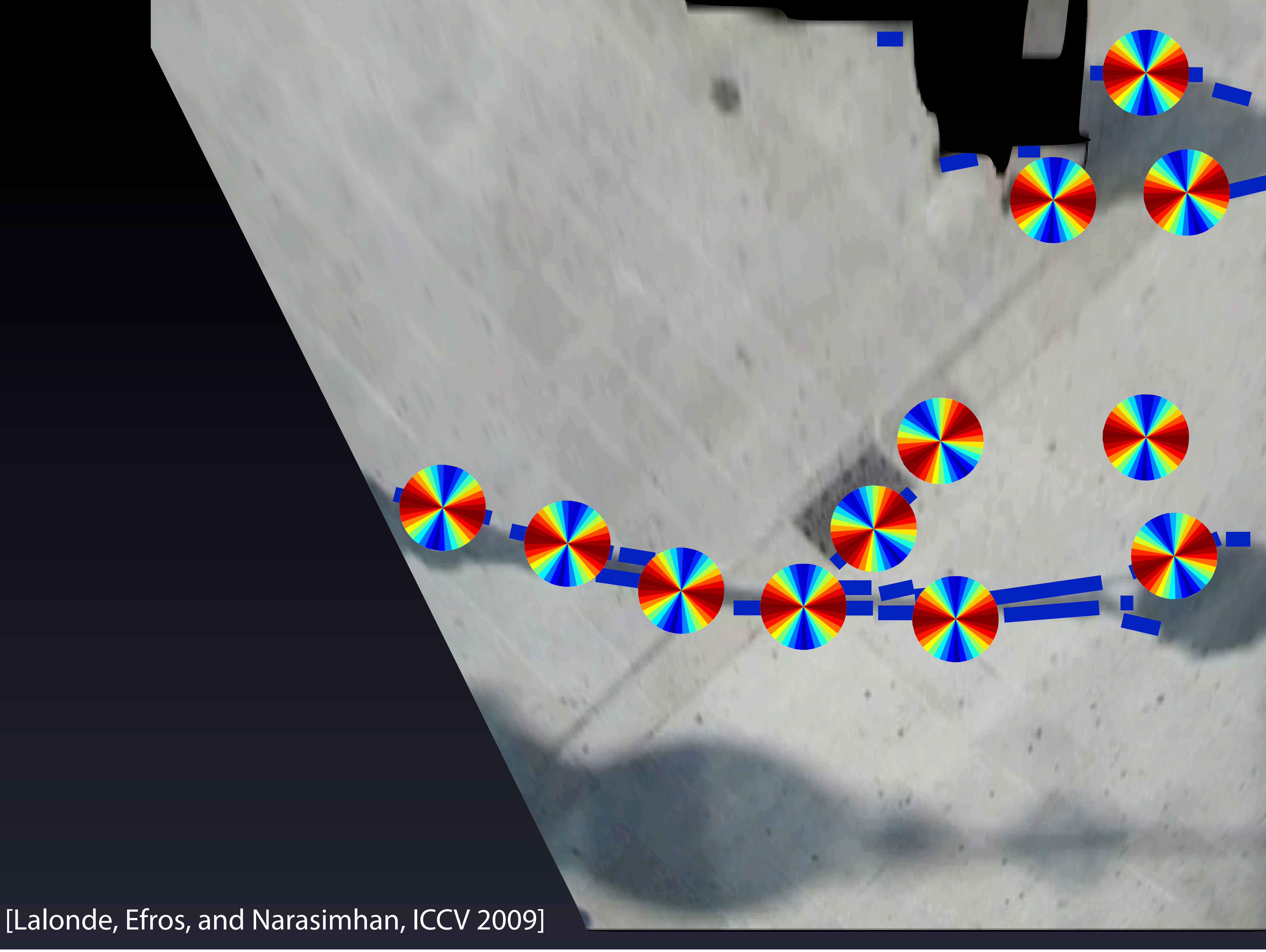
we model the sun azimuth distribution with 2 gaussians, separated by 180 degrees, since we have the directional ambiguity: the sun could either be here or here.



we model the sun azimuth distribution with 2 gaussians, separated by 180 degrees, since we have the directional ambiguity: the sun could either be here or here.



We do this for all the shadow lines, and combine their probability maps together in a voting scheme, which makes our approach robust to spurious edges.


[Lalonde, Efros, and Narasimhan, ICCV 2009]

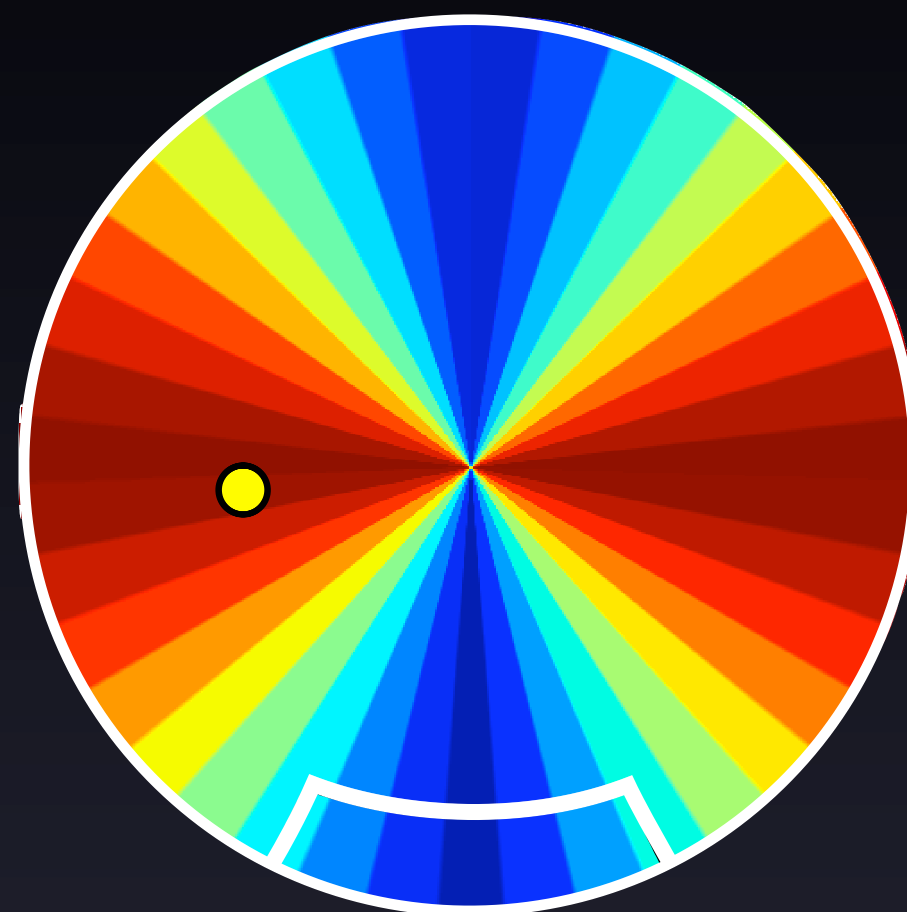
We do this for all the shadow lines, and combine their probability maps together in a voting scheme, which makes our approach robust to spurious edges.

P(sun azimuth | ground pixels)

[Lalonde, Efros, and Narasimhan, ICCV 2009]

This is the final result that we get. Using the other cues to resolve the directional ambiguity, we insert a virtual sun dial in the image and get very consistent results.

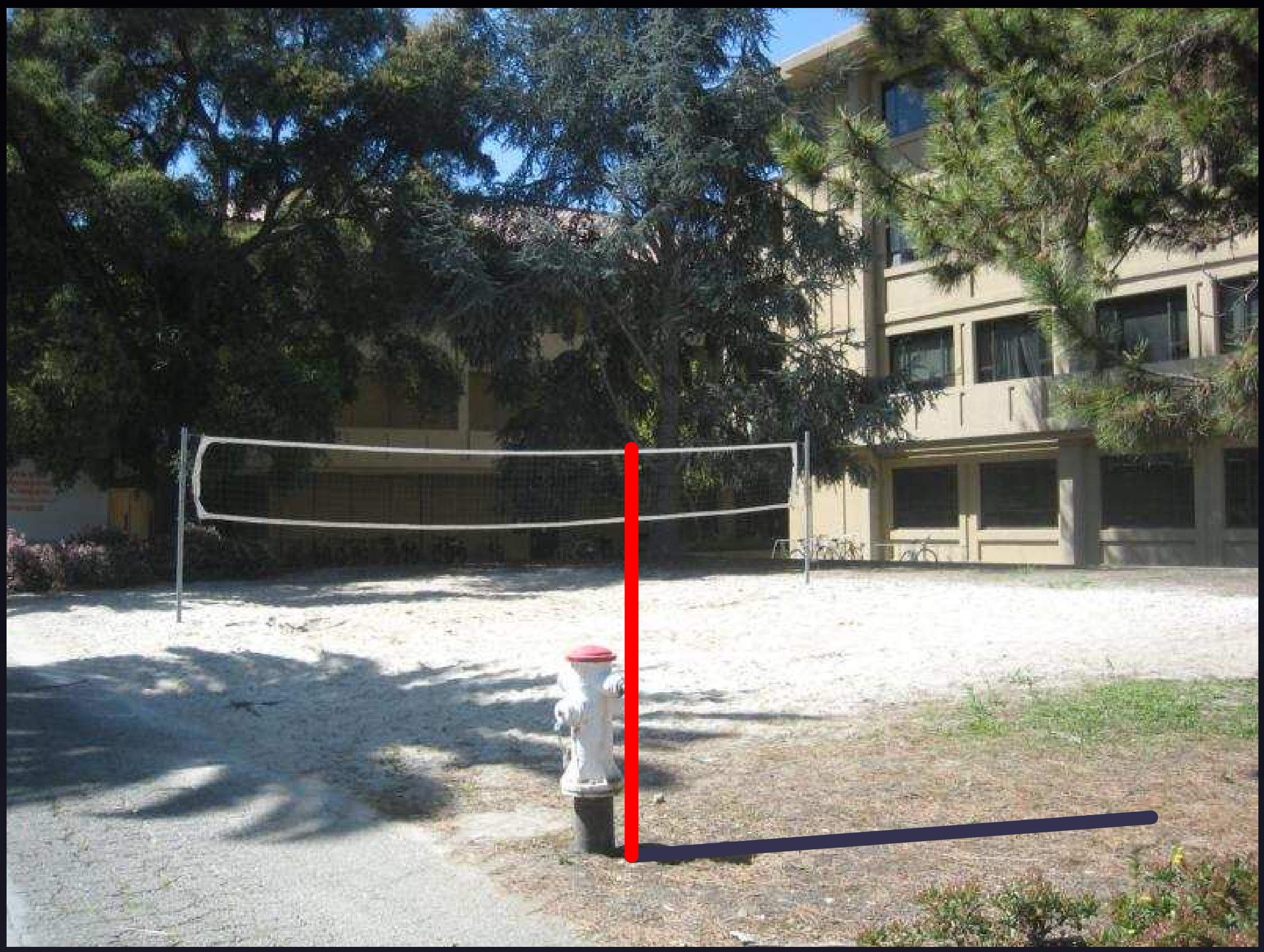
P(sun azimuth | ground pixels)

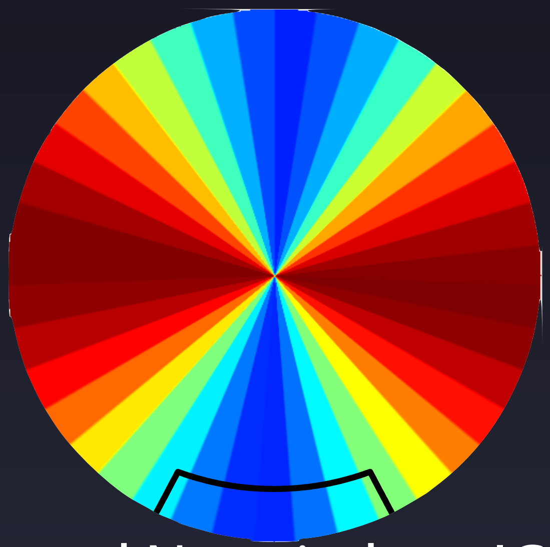


[Lalonde, Efros, and Narasimhan, ICCV 2009]

This is the final result that we get. Using the other cues to resolve the directional ambiguity, we insert a virtual sun dial in the image and get very consistent results.

Sun position given shadows

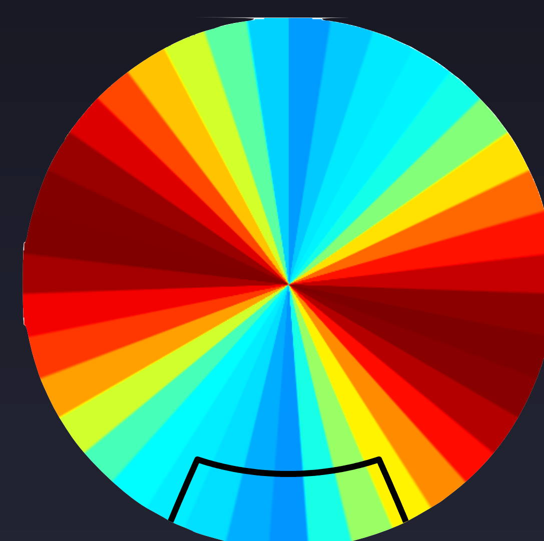




[Lalonde, Efros, and Narasimhan, ICCV 2009]

Here are two other examples.

Sun position given shadows

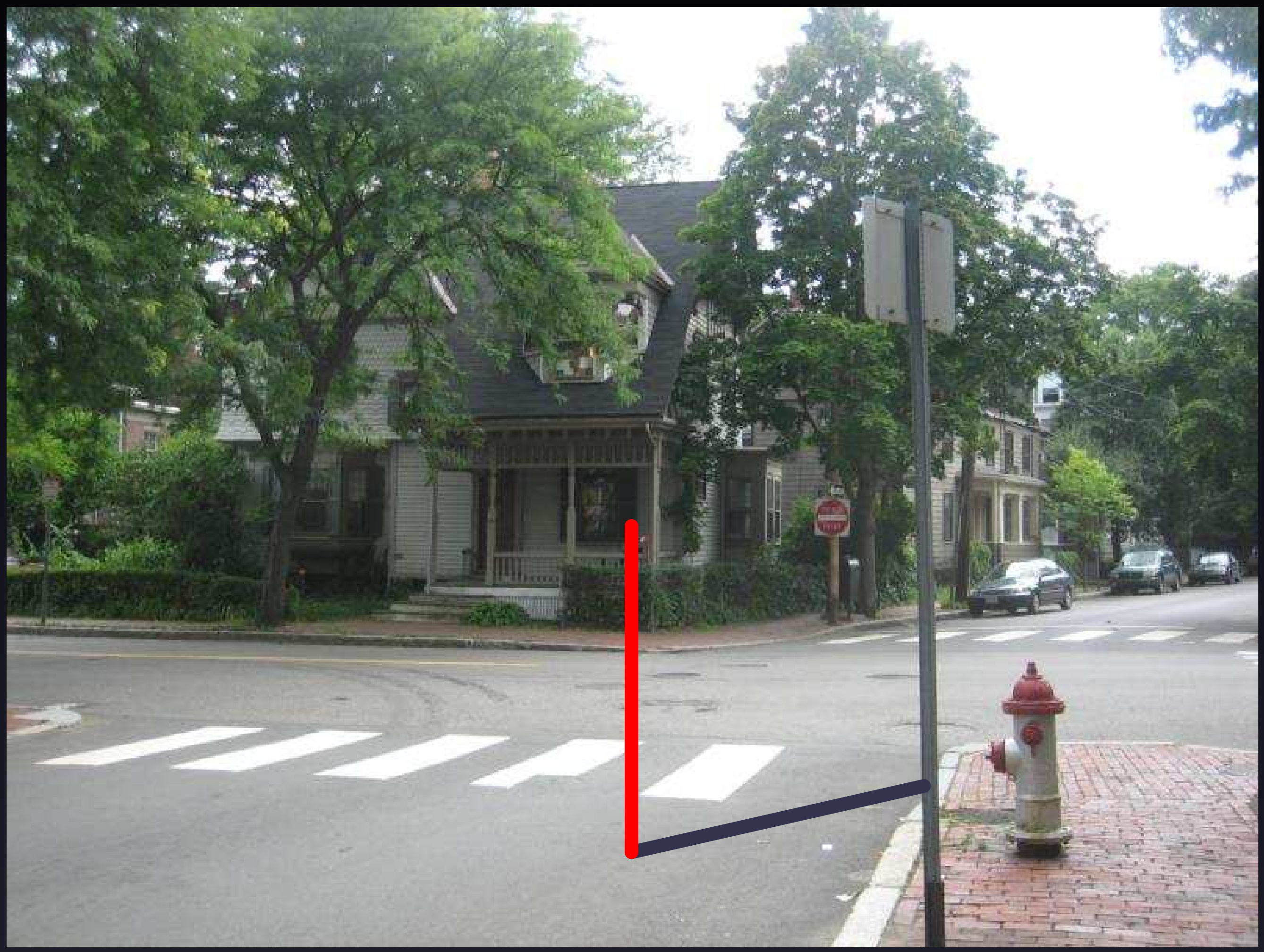


[Lalonde, Efros, and Narasimhan, ICCV 2009]

Here are two other examples.

Limitations of shadows cue

Shadow detection

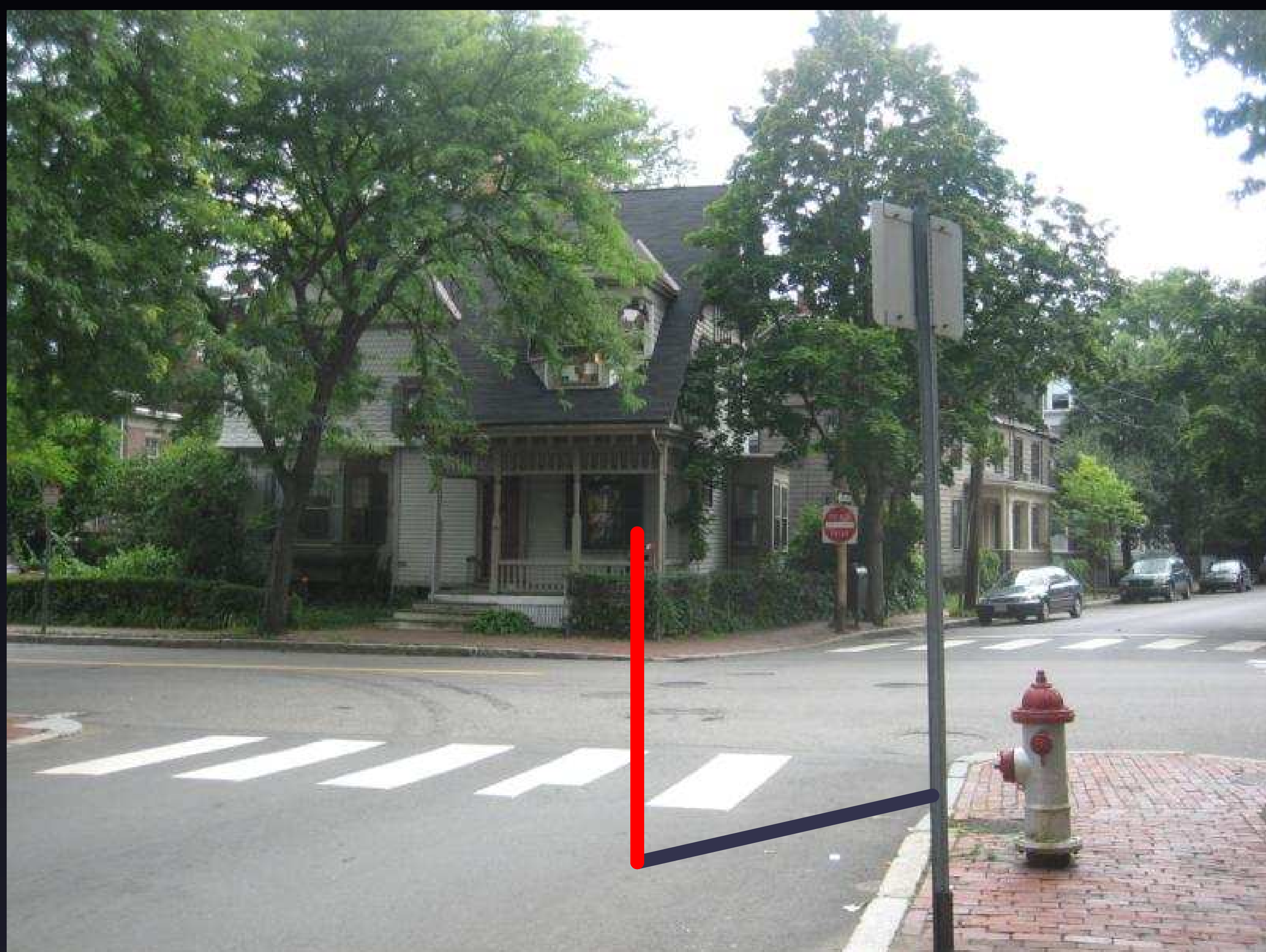


[Lalonde, Efros, and Narasimhan, ICCV 2009]

Our approach is sensitive to edges that look consistent throughout the image, but are not shadows, such as street markings.
The estimate can also be thrown off when shadows are not being cast by vertical objects.

Limitations of shadows cue

Shadow detection



Non-vertical objects

[Lalonde, Efros, and Narasimhan, ICCV 2009]

Our approach is sensitive to edges that look consistent throughout the image, but are not shadows, such as street markings.
The estimate can also be thrown off when shadows are not being cast by vertical objects.

[Lalonde, Efros, and Narasimhan, ICCV 2009]

Now that we've seen how we can exploit the shadows cast on the ground, let's see how we can use the shading on the vertical surfaces.

Sky

Ground

Vertical surfaces

[Lalonde, Efros, and Narasimhan, ICCV 2009]

Now that we’ve seen how we can exploit the shadows cast on the ground, let’s see how we can use the shading on the vertical surfaces.

[Lalonde, Efros, and Narasimhan, ICCV 2009]

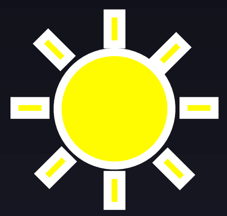
Now that we've seen how we can exploit the shadows cast on the ground, let's see how we can use the shading on the vertical surfaces.

Surfaces

[Lalonde, Efros, and Narasimhan, ICCV 2009]

When looking at such an image, we know that the sun comes from the left since this surface is brighter than this one. In order to compute that, we need to somehow extract geometry from the image.

Surfaces



[Lalonde, Efros, and Narasimhan, ICCV 2009]

When looking at such an image, we know that the sun comes from the left since this surface is brighter than this one. In order to compute that, we need to somehow extract geometry from the image.

Surfaces

[Lalonde, Efros, and Narasimhan, ICCV 2009]

When looking at such an image, we know that the sun comes from the left since this surface is brighter than this one. In order to compute that, we need to somehow extract geometry from the image.

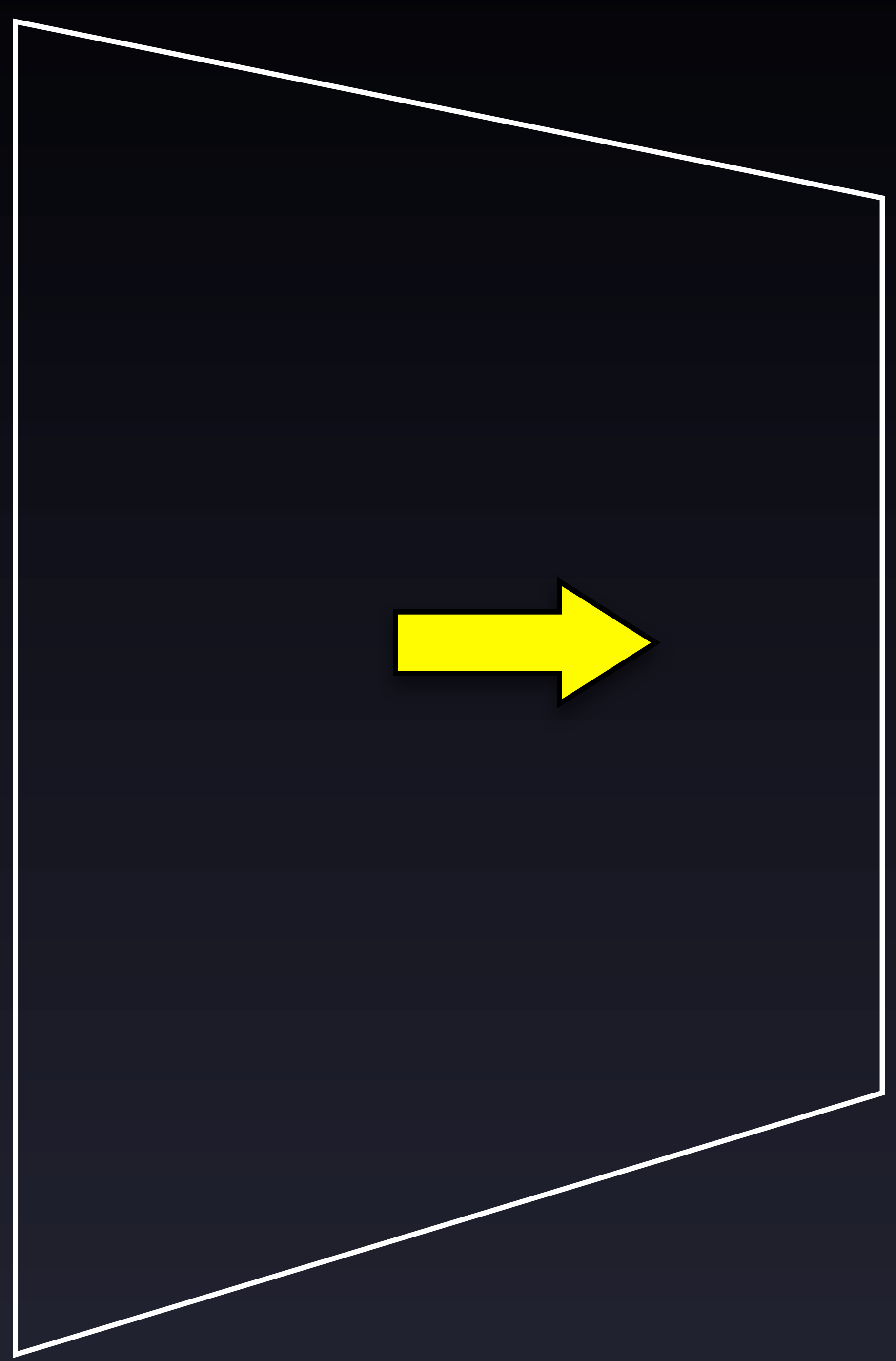
Vertical surfaces

[Lalonde, Efros, and Narasimhan, ICCV 2009]

[Hoiem *et al.*, IJCV '07]

For this, we again use the geometric context algorithm of Hoiem et al. to split the vertical surfaces into 3 groups: facing right, facing left, and facing towards the camera.
The idea is that each surface predicts a sun position in the direction of its normal, and we combine them together by weighting them according to their relative brightness. Note that this is not reliable enough to predict the sun elevation angle, so we focus on the azimuth only.
Now this only works if the albedos are either the same, or known, neither of which is true.

Vertical surfaces

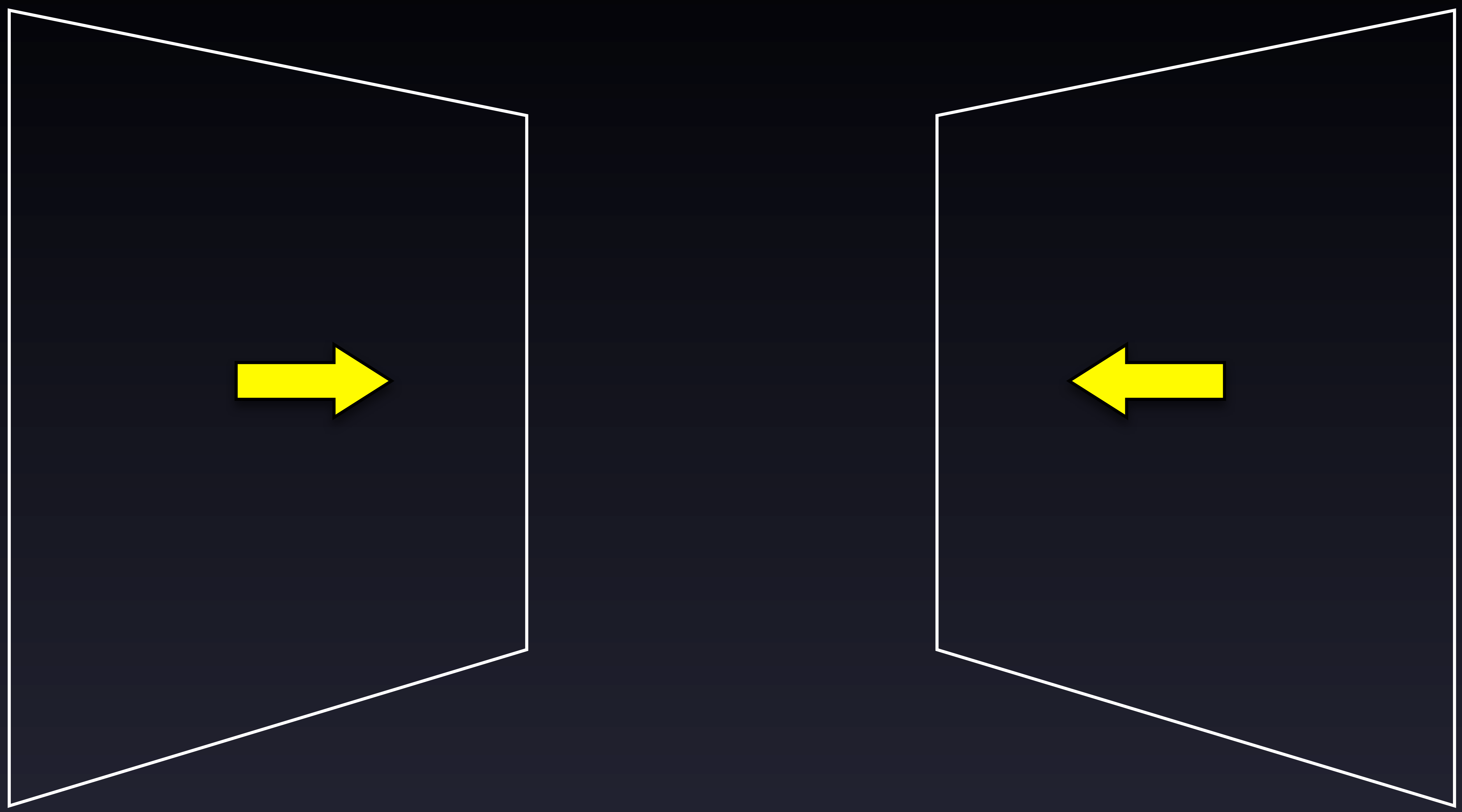


[Lalonde, Efros, and Narasimhan, ICCV 2009]

[Hoiem *et al.*, IJCV '07]

For this, we again use the geometric context algorithm of Hoiem et al. to split the vertical surfaces into 3 groups: facing right, facing left, and facing towards the camera.
The idea is that each surface predicts a sun position in the direction of its normal, and we combine them together by weighting them according to their relative brightness. Note that this is not reliable enough to predict the sun elevation angle, so we focus on the azimuth only.
Now this only works if the albedos are either the same, or known, neither of which is true.

Vertical surfaces

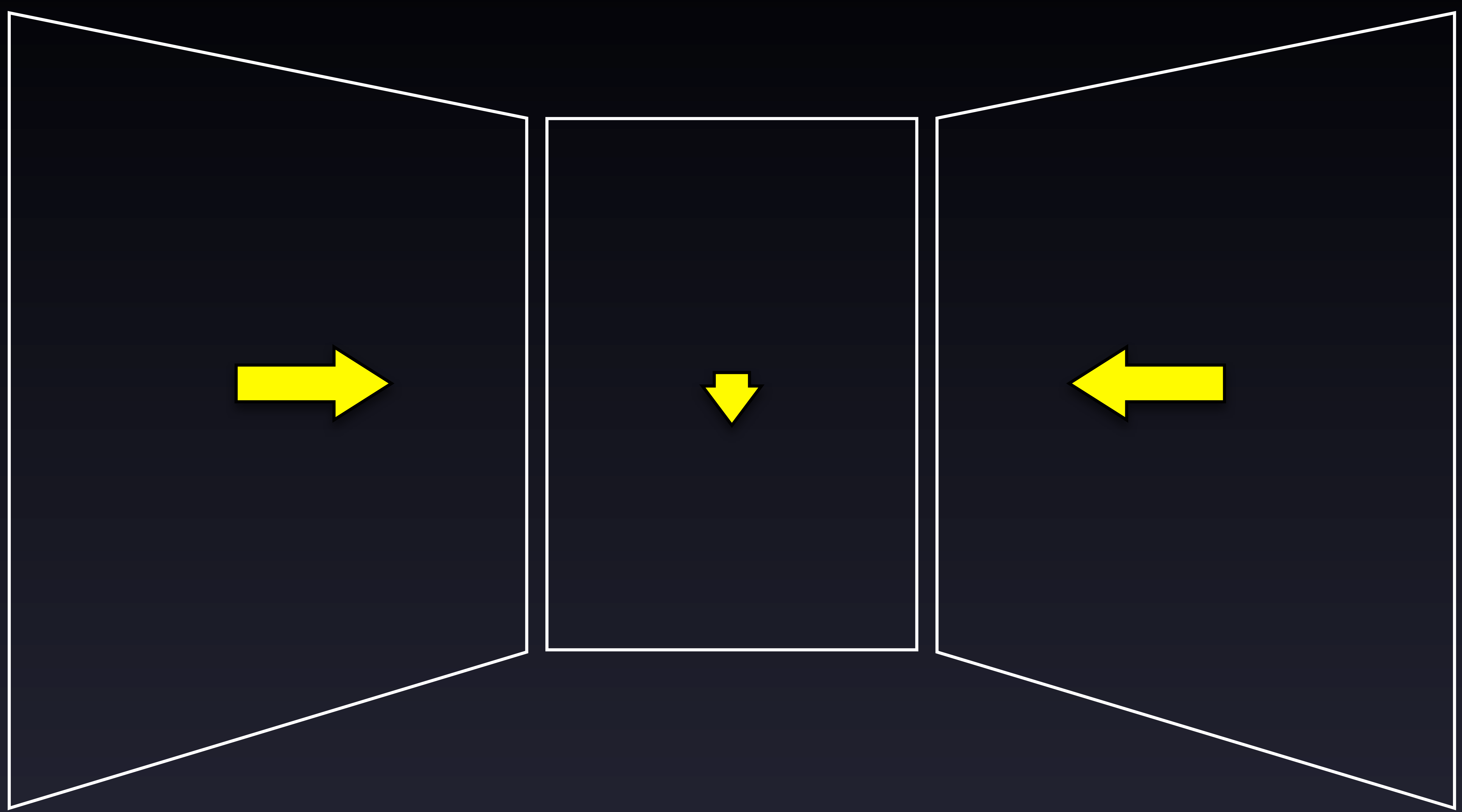


[Lalonde, Efros, and Narasimhan, ICCV 2009]

[Hoiem *et al.*, IJCV '07]

For this, we again use the geometric context algorithm of Hoiem et al. to split the vertical surfaces into 3 groups: facing right, facing left, and facing towards the camera.
The idea is that each surface predicts a sun position in the direction of its normal, and we combine them together by weighting them according to their relative brightness. Note that this is not reliable enough to predict the sun elevation angle, so we focus on the azimuth only.
Now this only works if the albedos are either the same, or known, neither of which is true.

Vertical surfaces

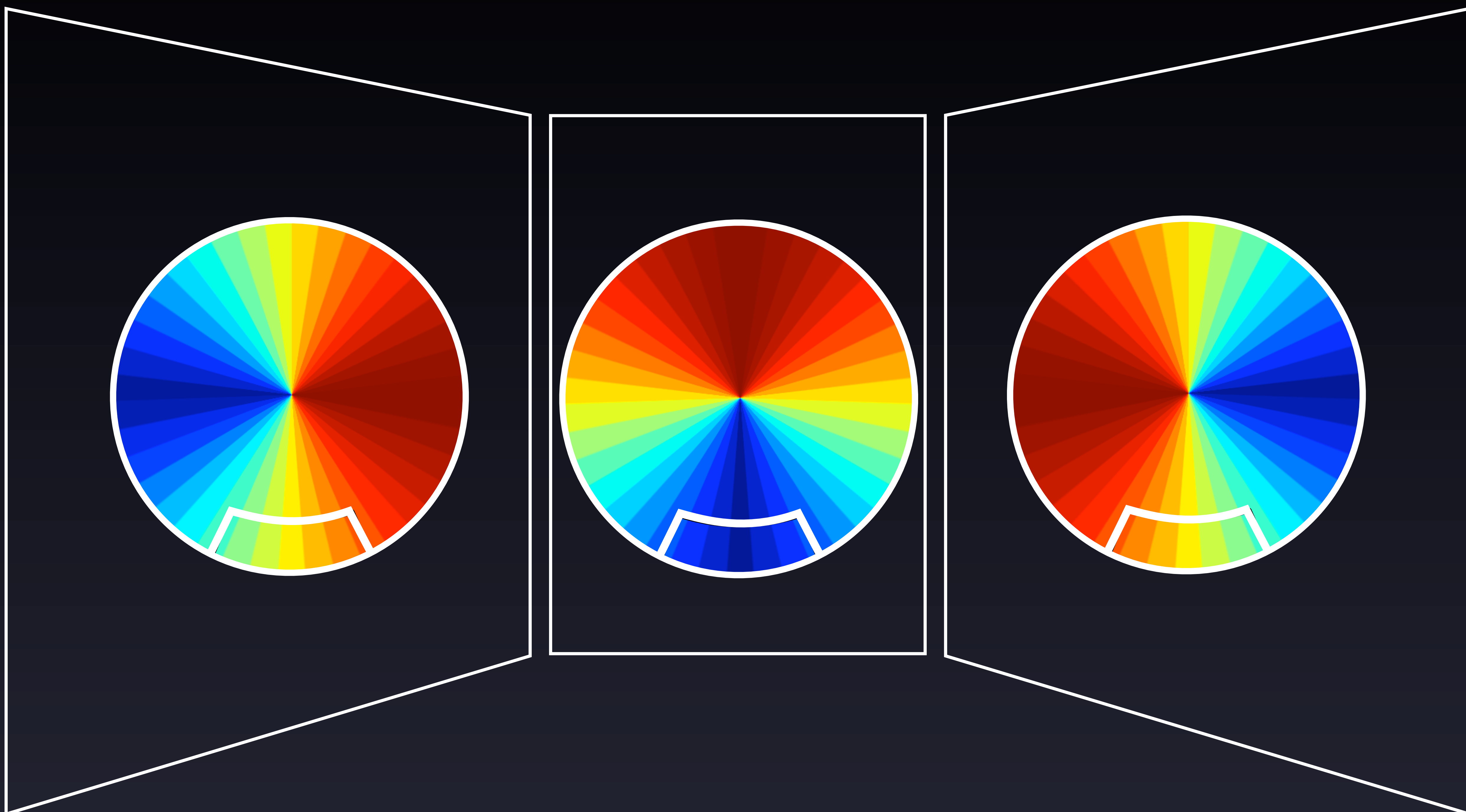


[Lalonde, Efros, and Narasimhan, ICCV 2009]

[Hoiem *et al.*, IJCV '07]

For this, we again use the geometric context algorithm of Hoiem et al. to split the vertical surfaces into 3 groups: facing right, facing left, and facing towards the camera.
The idea is that each surface predicts a sun position in the direction of its normal, and we combine them together by weighting them according to their relative brightness. Note that this is not reliable enough to predict the sun elevation angle, so we focus on the azimuth only.
Now this only works if the albedos are either the same, or known, neither of which is true.

Vertical surfaces



[Lalonde, Efros, and Narasimhan, ICCV 2009] [Hoiem *et al.*, IJCV ’07]

For this, we again use the geometric context algorithm of Hoiem et al. to split the vertical surfaces into 3 groups: facing right, facing left, and facing towards the camera.
The idea is that each surface predicts a sun position in the direction of its normal, and we combine them together by weighting them according to their relative brightness. Note that this is not reliable enough to predict the sun elevation angle, so we focus on the azimuth only.
Now this only works if the albedos are either the same, or known, neither of which is true.

Surfaces

[Lalonde, Efros, and Narasimhan, ICCV 2009]

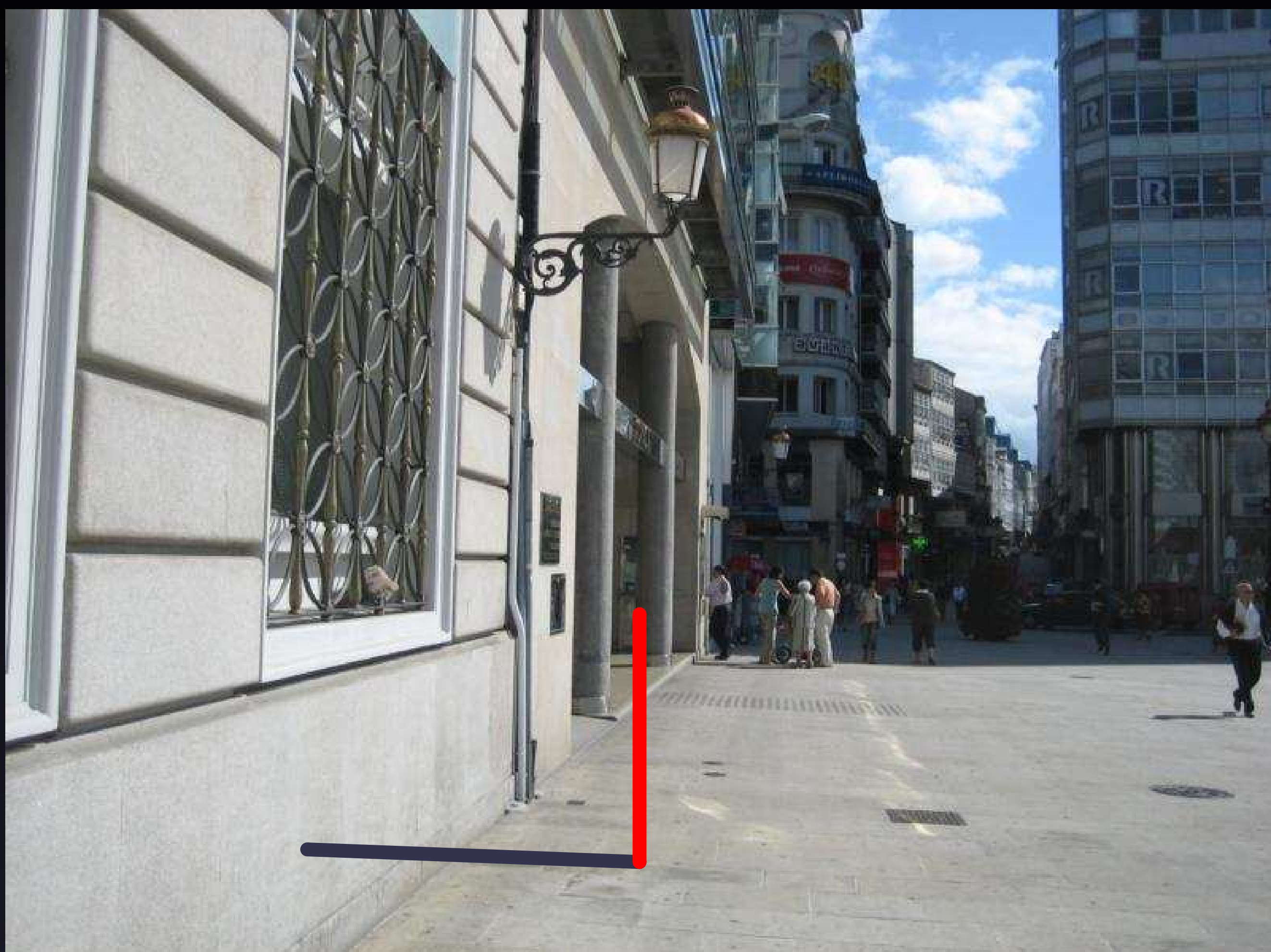
But look at the image again. When the sun shines directly on a surface, the effects due to illumination overwhelm those due to albedo, so we can actually compare their brightnesses directly.

Surfaces

[Lalonde, Efros, and Narasimhan, ICCV 2009]

But look at the image again. When the sun shines directly on a surface, the effects due to illumination overwhelm those due to albedo, so we can actually compare their brightnesses directly.

Sun position given surfaces



[Lalonde, Efros, and Narasimhan, ICCV 2009]

This cue is definitely the weakest of all 3, since the albedo assumption, the vertical surface classification and complex cast shadows can affect its output. But we found that most of the time, it’s still useful to figure out the rough sun direction, and helps resolve the shadow ambiguity as in the examples shown here.

Sun position given surfaces

[Lalonde, Efros, and Narasimhan, ICCV 2009]

This cue is definitely the weakest of all 3, since the albedo assumption, the vertical surface classification and complex cast shadows can affect its output. But we found that most of the time, it's still useful to figure out the rough sun direction, and helps resolve the shadow ambiguity as in the examples shown here.

Limitations of surfaces cue

[Lalonde, Efros, and Narasimhan, ICCV 2009]

Limitations of surfaces cue

No flat surfaces

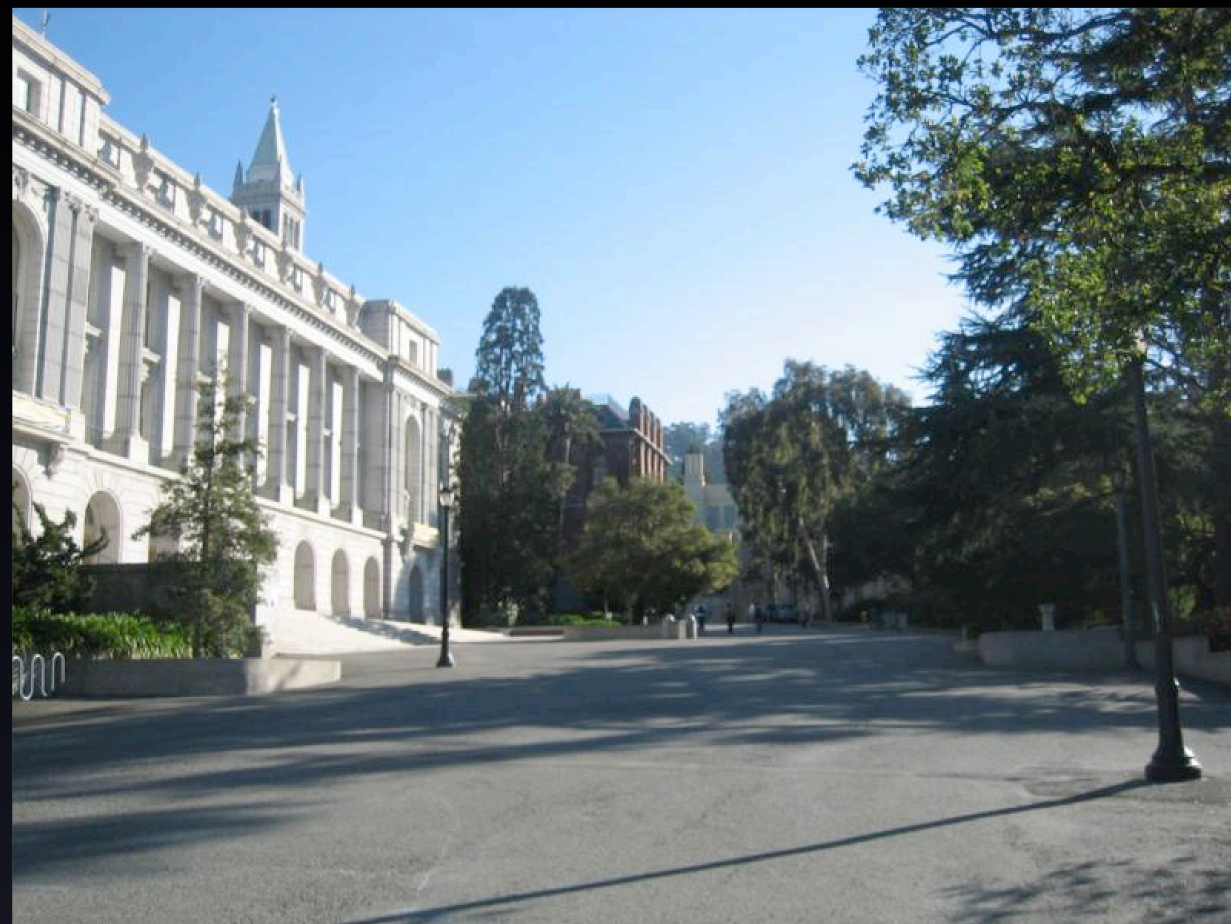
[Lalonde, Efros, and Narasimhan, ICCV 2009]

Cue combination

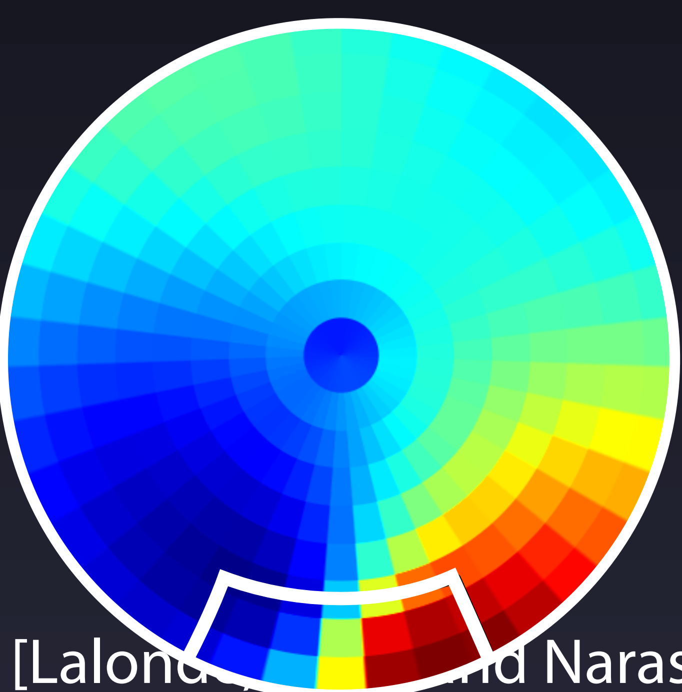
[Lalonde, Efros, and Narasimhan, ICCV 2009]

Now that we've computed the 3 scene cues: probability of the sun position given the sky, shadows, and vertical surfaces, we can combine them with our prior on the sun elevation to obtain a final, more robust estimate of the sun position distribution.

Cue combination



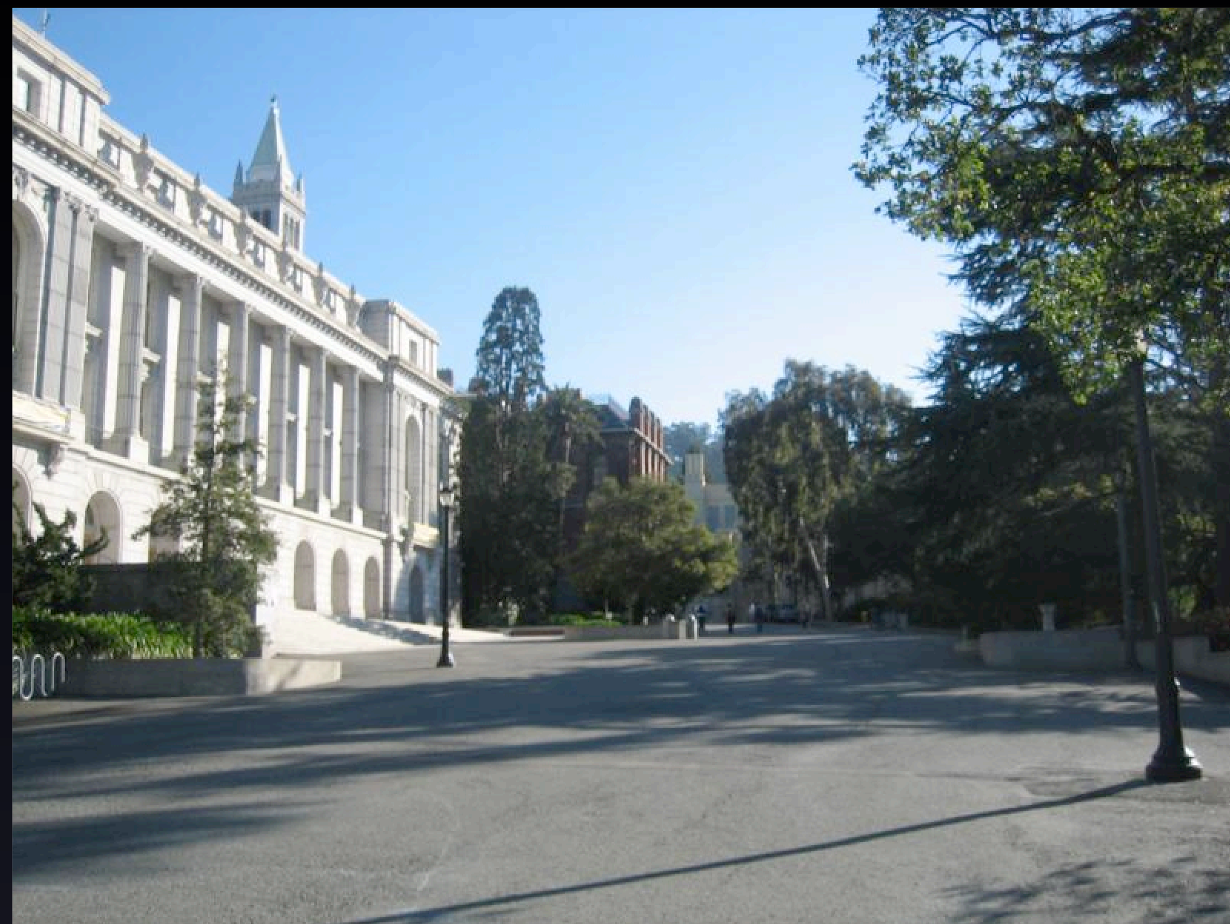
P(sun position | sky)



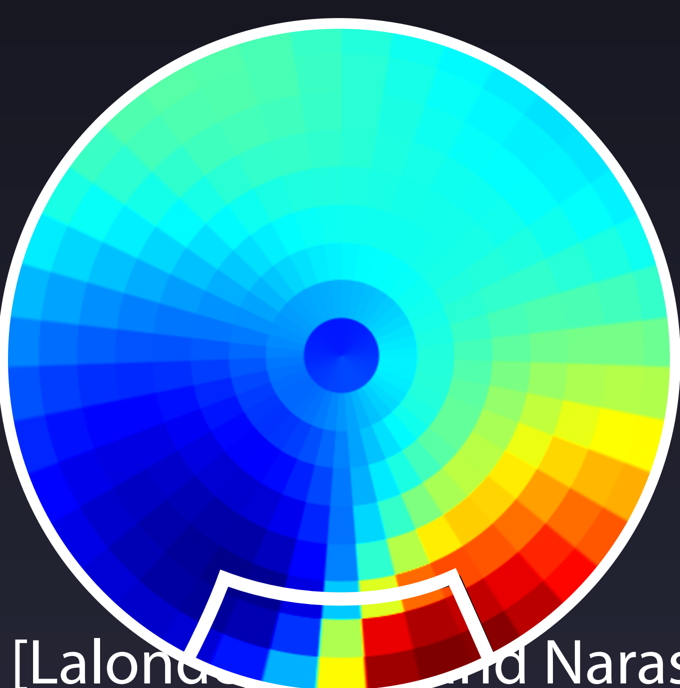
[Lalond nd Narasimhan, ICCV 2009]

Now that we've computed the 3 scene cues: probability of the sun position given the sky, shadows, and vertical surfaces, we can combine them with our prior on the sun elevation to obtain a final, more robust estimate of the sun position distribution.

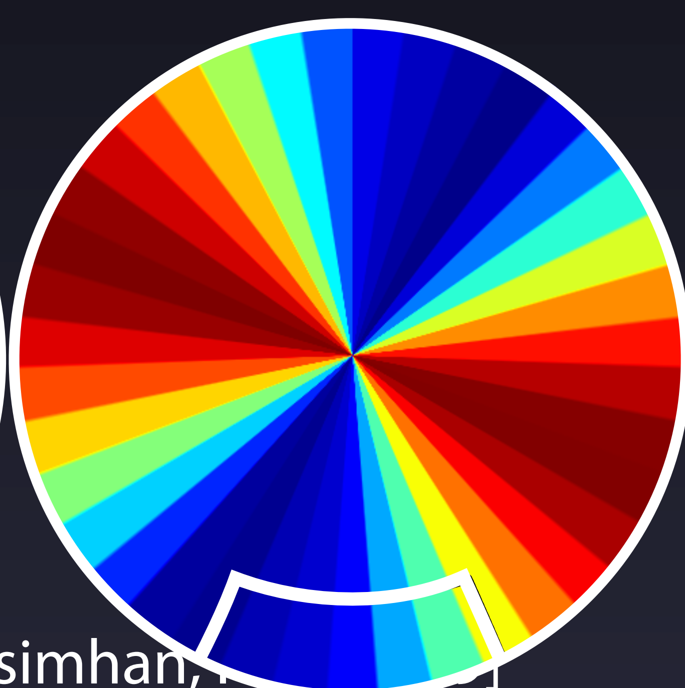
Cue combination



P(sun position | sky)



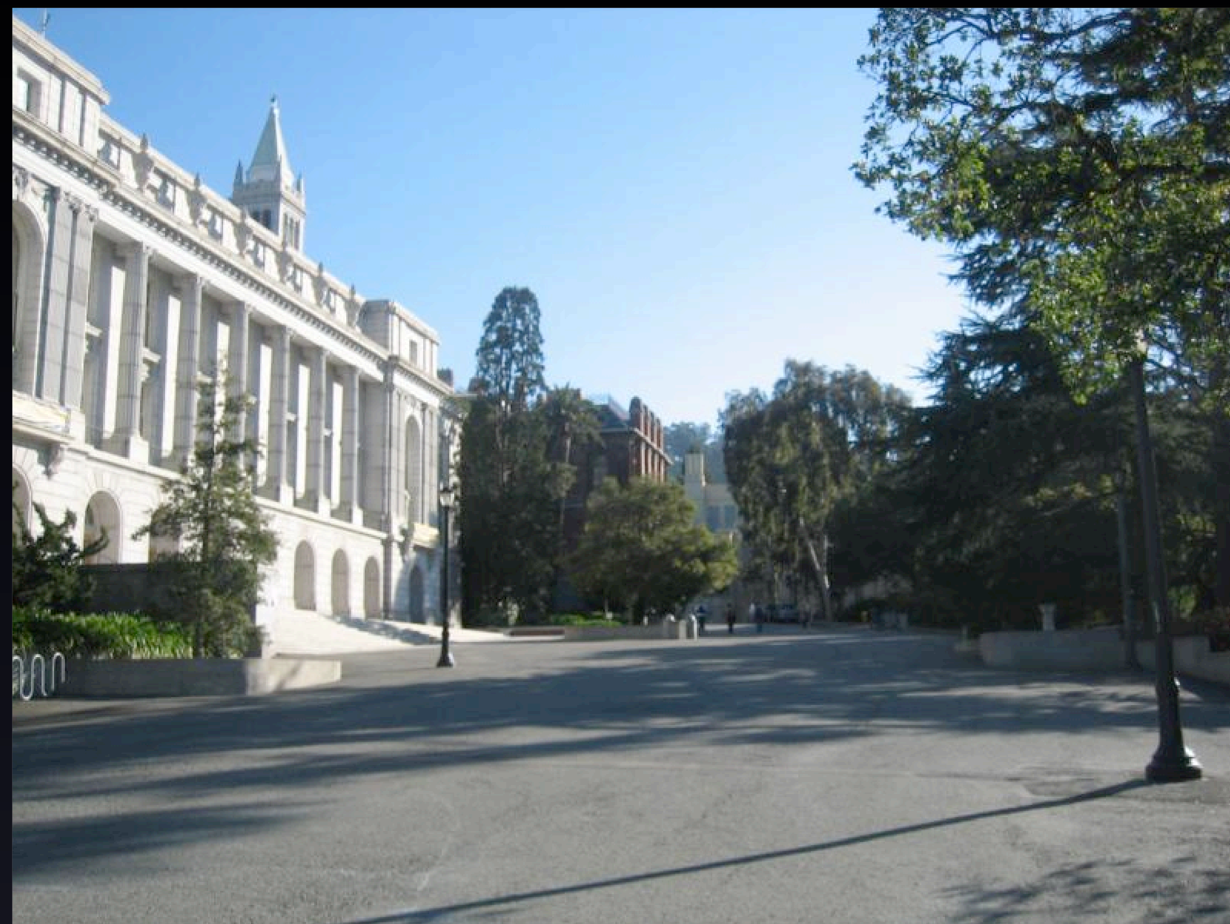
P(sun position | shadows)



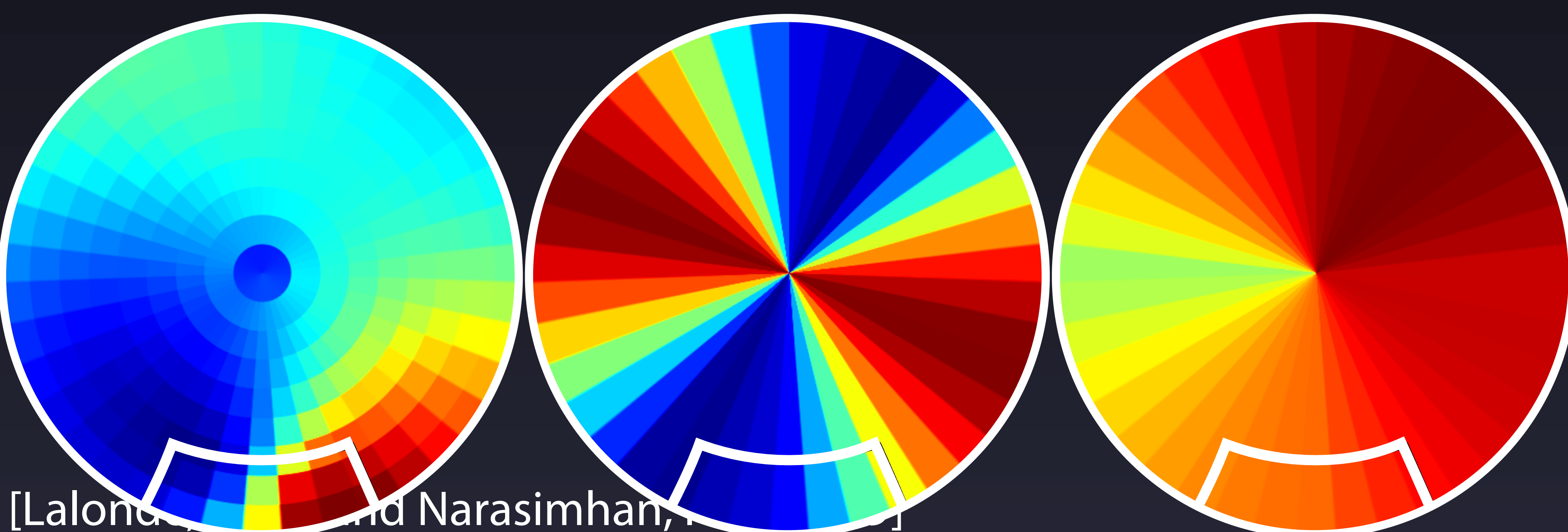
[Lalond… nd Narasimhan, …]

Now that we've computed the 3 scene cues: probability of the sun position given the sky, shadows, and vertical surfaces, we can combine them with our prior on the sun elevation to obtain a final, more robust estimate of the sun position distribution.

Cue combination



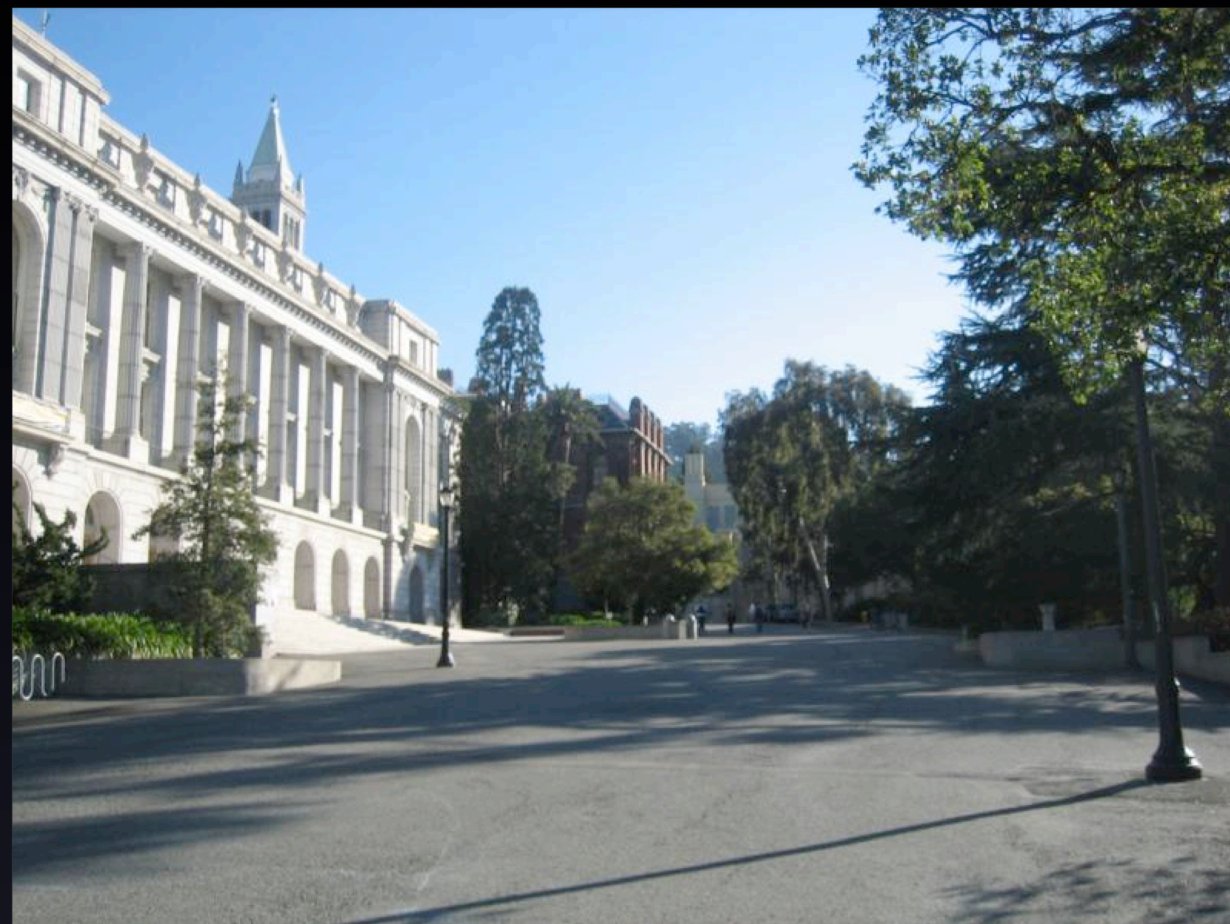
P(sun position | sky) P(sun position | shadows) P(sun position | surfaces)



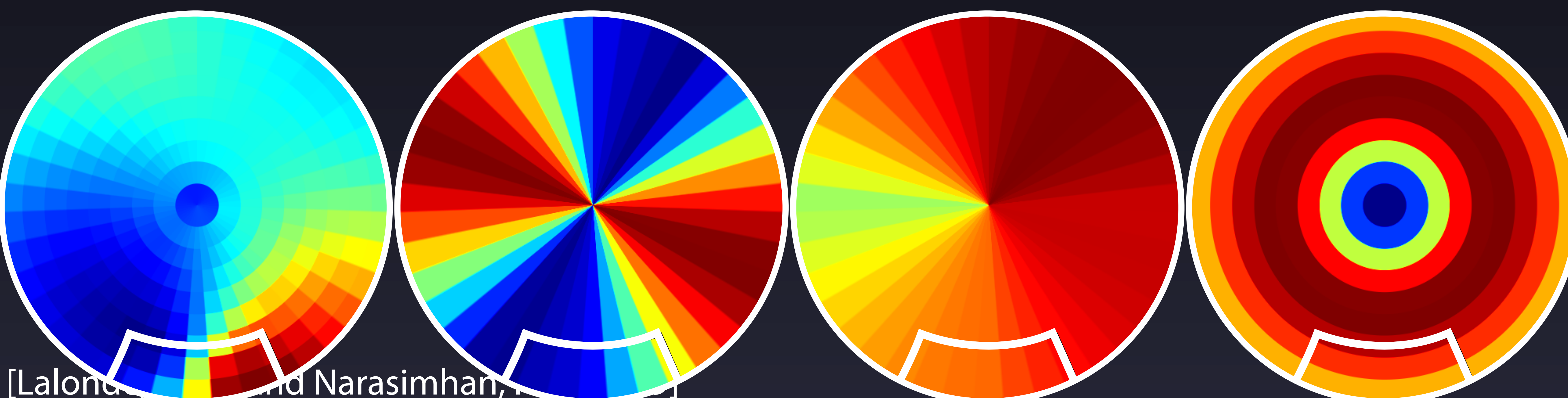
[Lalond... nd Narasimhan, ...]

Now that we've computed the 3 scene cues: probability of the sun position given the sky, shadows, and vertical surfaces, we can combine them with our prior on the sun elevation to obtain a final, more robust estimate of the sun position distribution.

Cue combination



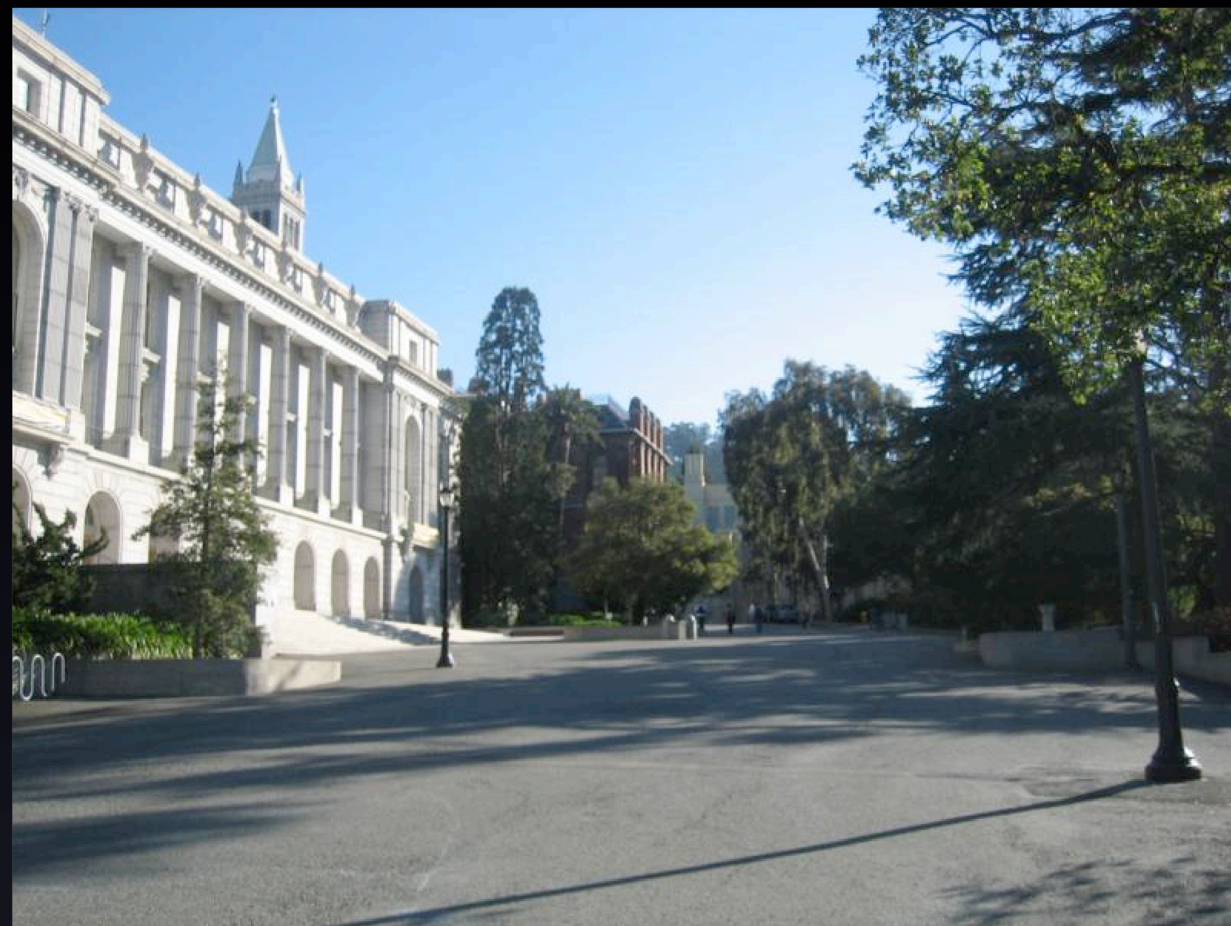
P(sun position | sky) P(sun position | shadows) P(sun position | surfaces) P(sun position)



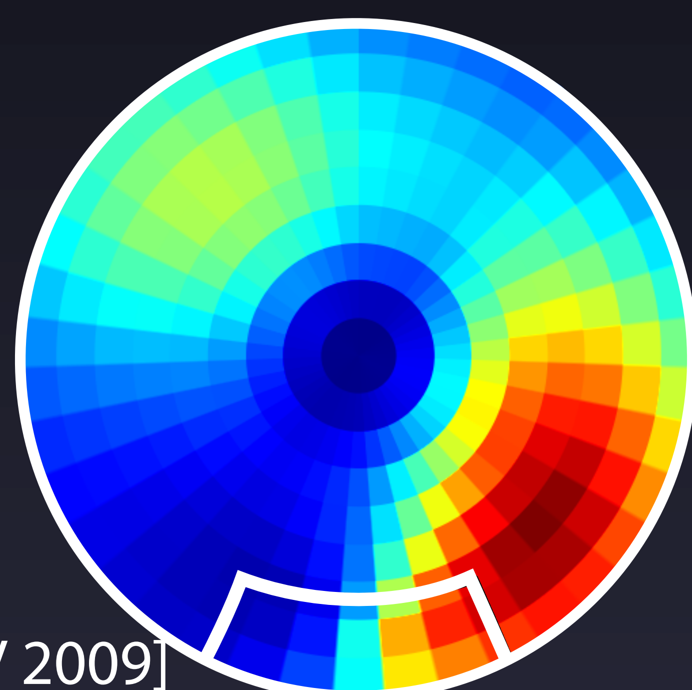
[Lalond nd Narasimhan,]

Now that we've computed the 3 scene cues: probability of the sun position given the sky, shadows, and vertical surfaces, we can combine them with our prior on the sun elevation to obtain a final, more robust estimate of the sun position distribution.

Cue combination



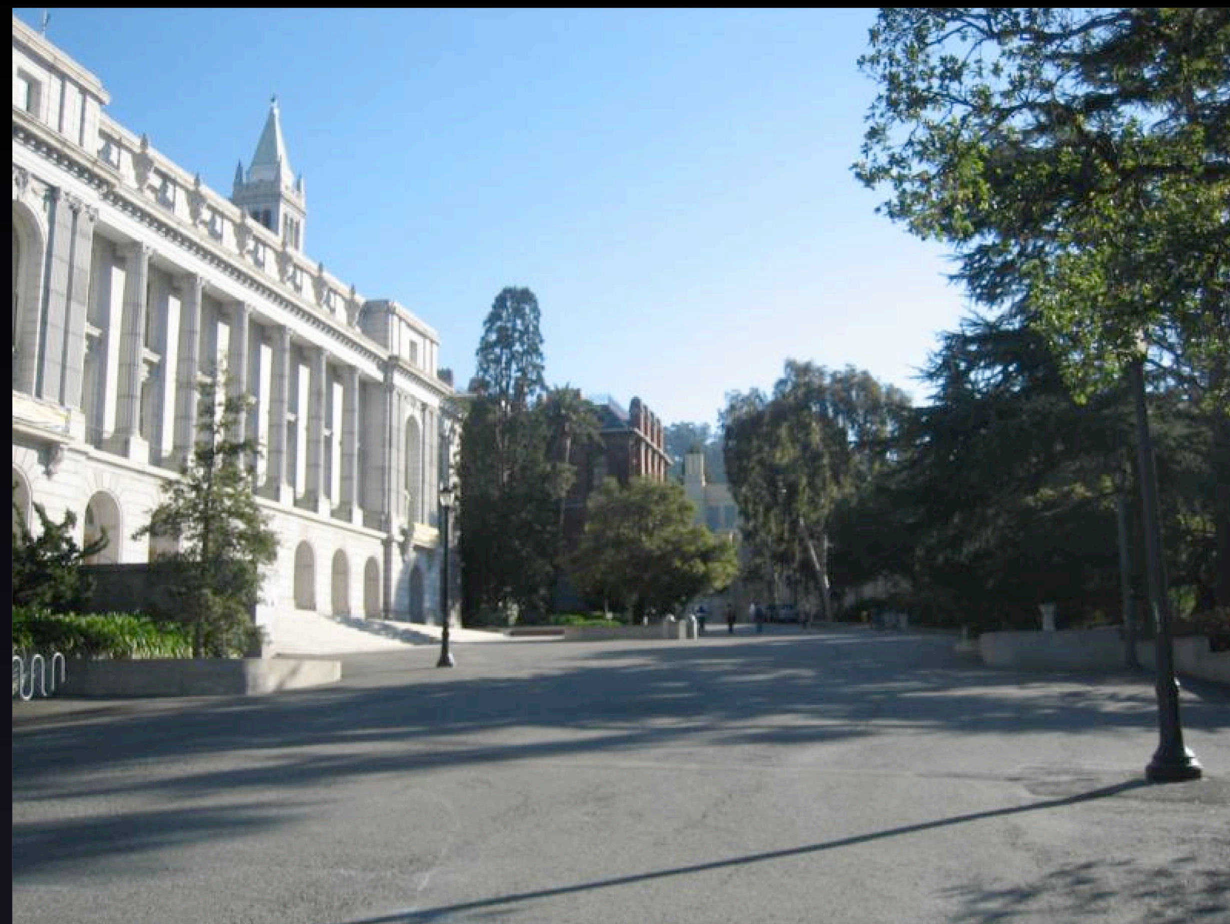
P(sun position | image)



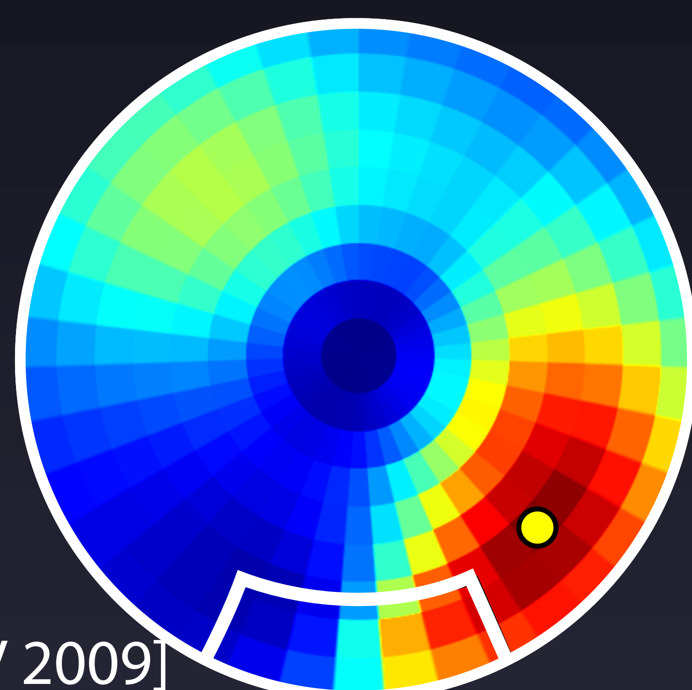
[Lalonde, Efros, and Narasimhan, ICCV 2009]

Now that we've computed the 3 scene cues: probability of the sun position given the sky, shadows, and vertical surfaces, we can combine them with our prior on the sun elevation to obtain a final, more robust estimate of the sun position distribution.

Cue combination



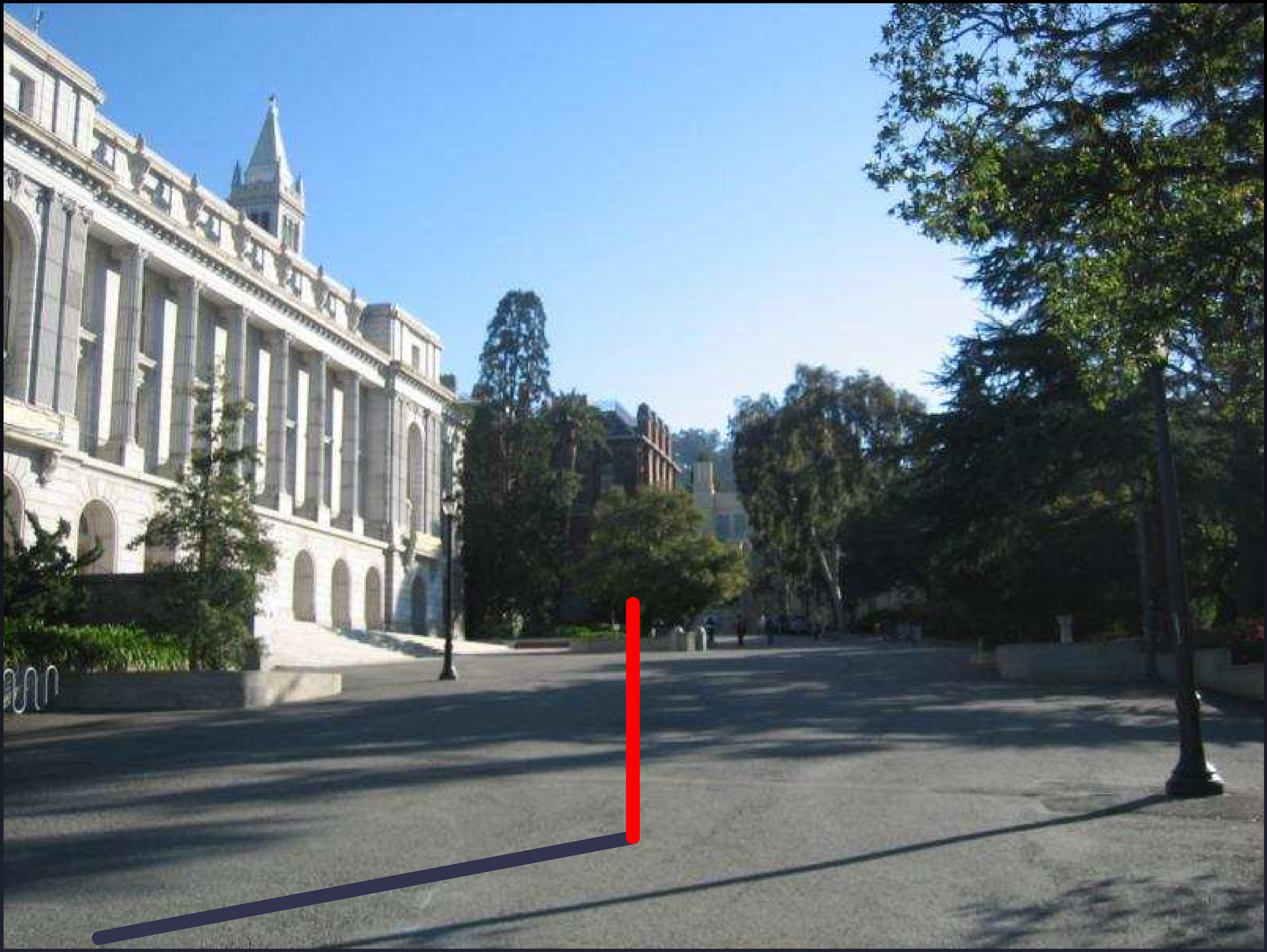
P(sun position | image)



[Lalonde, Efros, and Narasimhan, ICCV 2009]

Now that we've computed the 3 scene cues: probability of the sun position given the sky, shadows, and vertical surfaces, we can combine them with our prior on the sun elevation to obtain a final, more robust estimate of the sun position distribution.

Cue combination



[Lalonde, Efros, and Narasimhan, ICCV 2009]

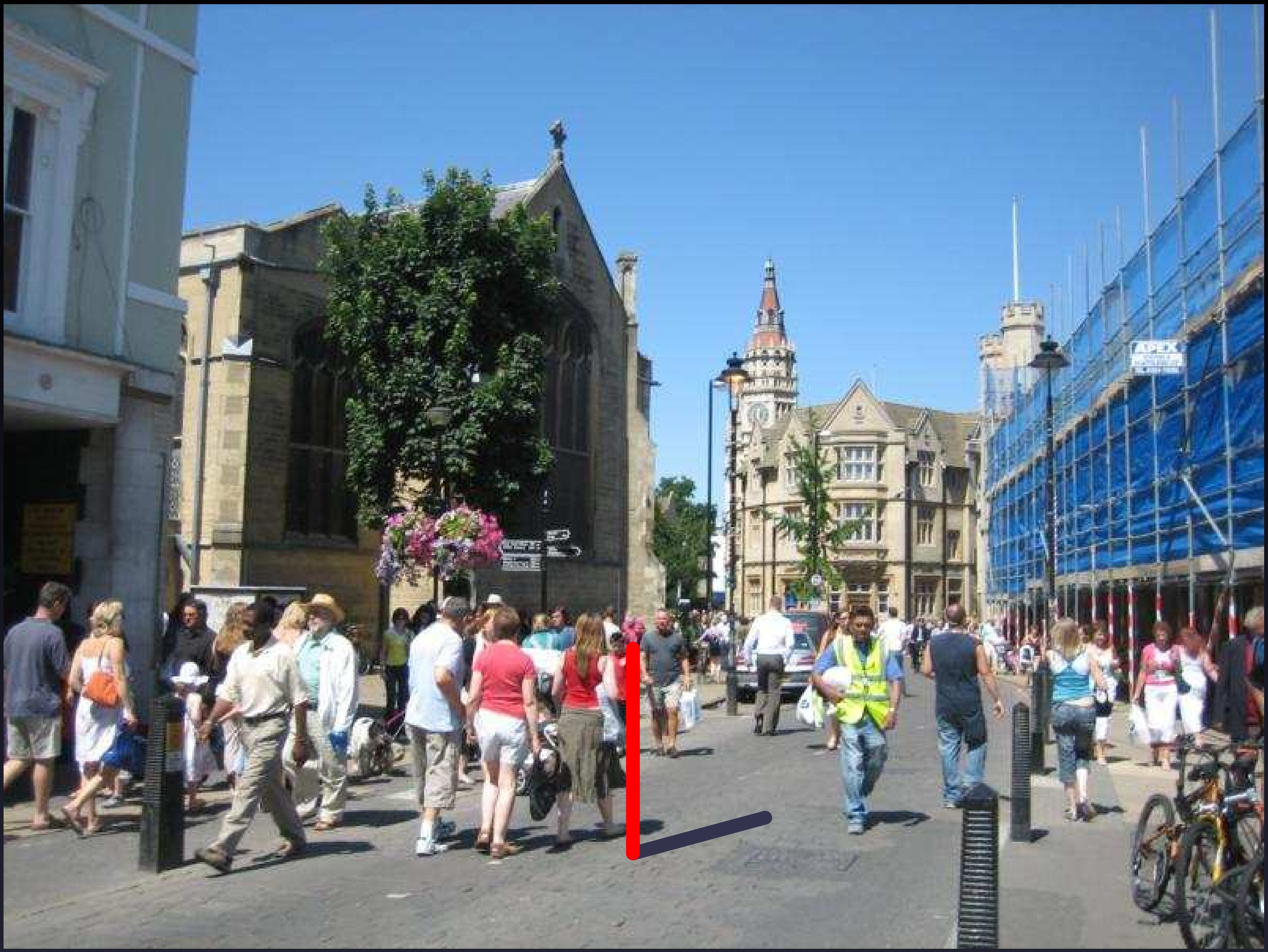
Now that we've computed the 3 scene cues: probability of the sun position given the sky, shadows, and vertical surfaces, we can combine them with our prior on the sun elevation to obtain a final, more robust estimate of the sun position distribution.

[Lalonde, Efros, and Narasimhan, ICCV 2009]

Here are a few qualitative results. Here's an example where the shadows are the most important cue.

[Lalonde, Efros, and Narasimhan, ICCV 2009]

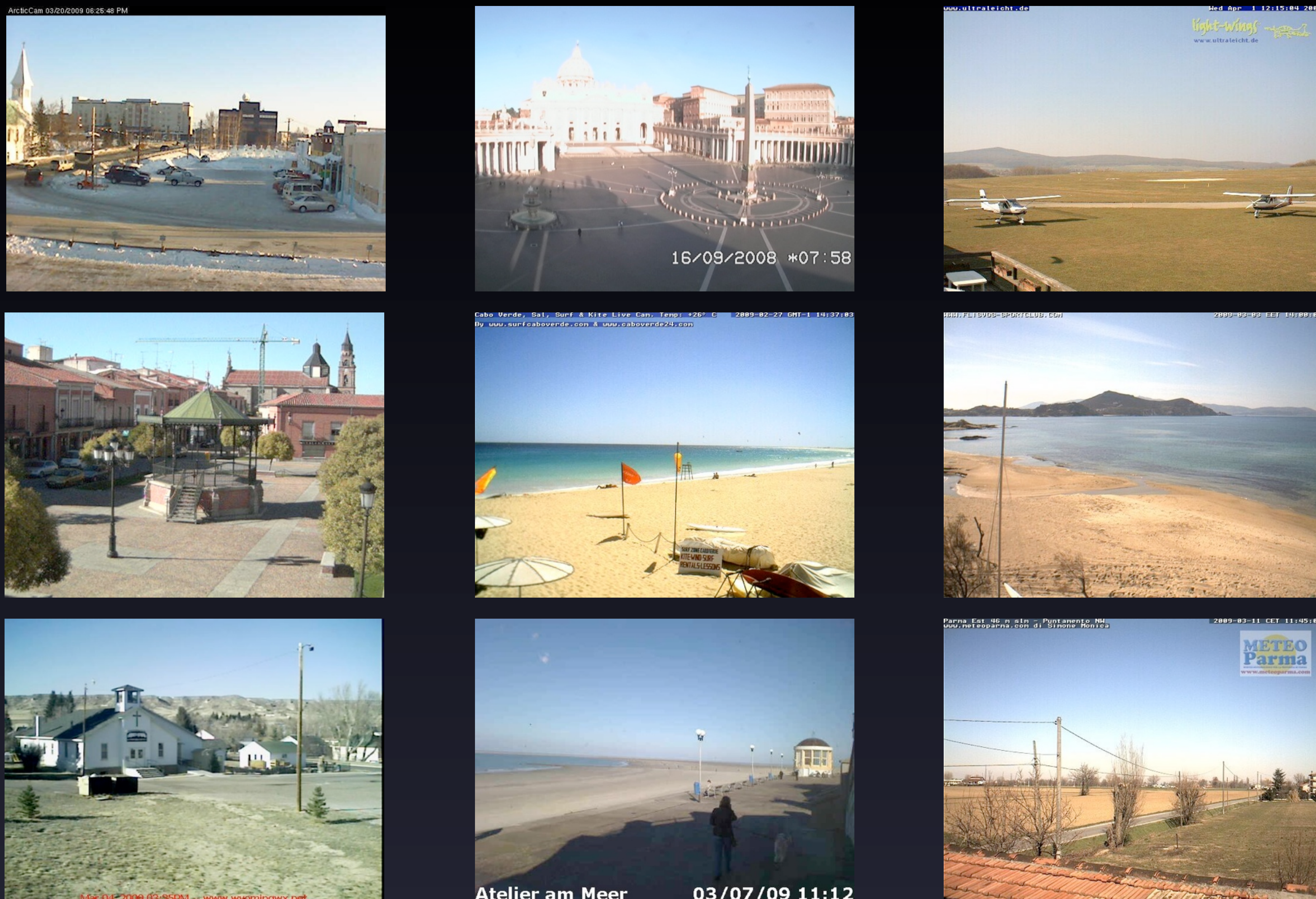
Shadows and vertical surfaces, more clutter



[Lalonde, Efros, and Narasimhan, ICCV 2009]

Here's a challenging example, full of clutter

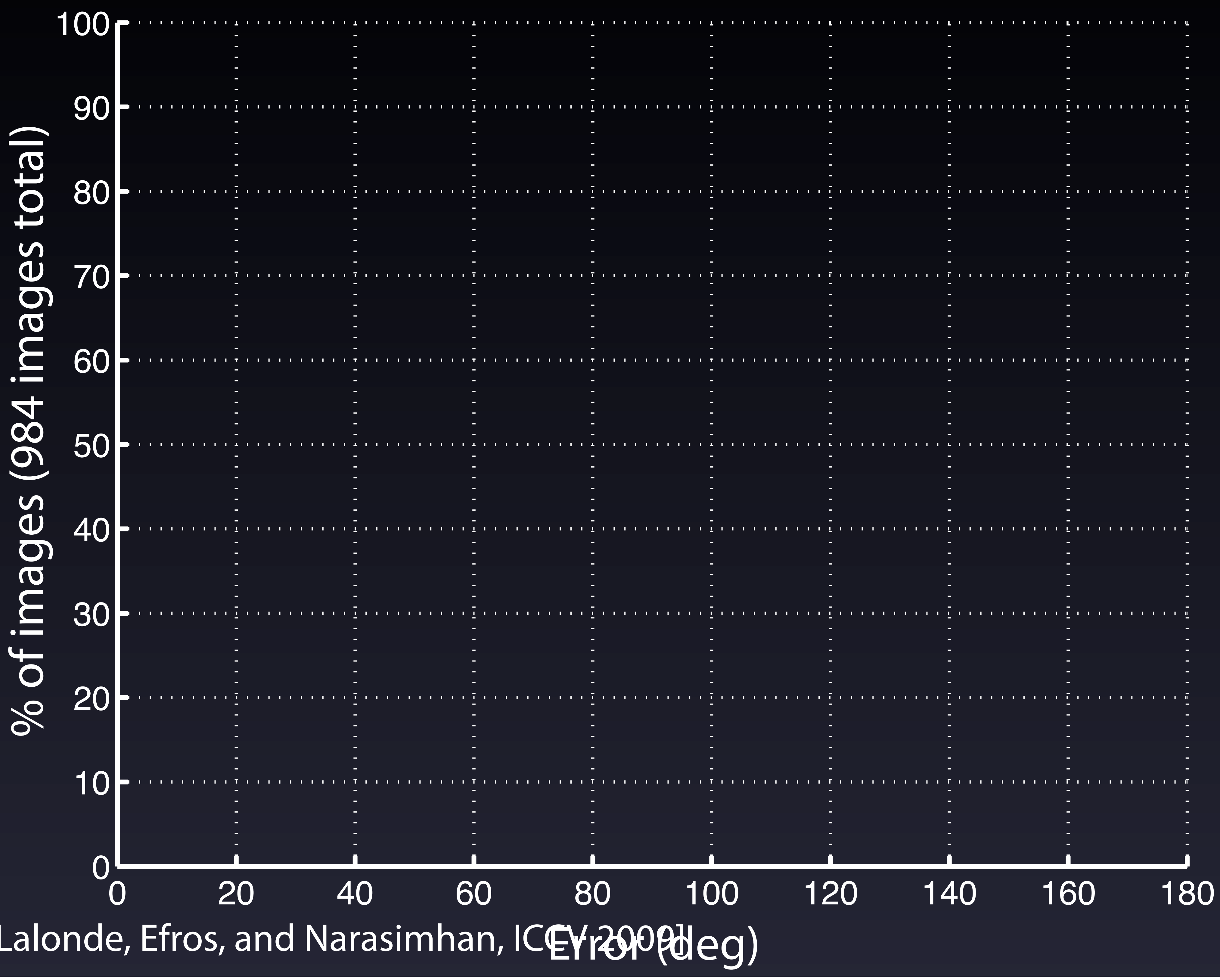
Quantitative evaluation



[Lalonde, Efros, and Narasimhan, ICCV 2009]

We also performed a quantitative evaluation of our approach, on more than 950 images taken from 15 different calibrated webcams, of which I show an example here. At each frame, we know where the sun is with respect to the camera.

Quantitative evaluation

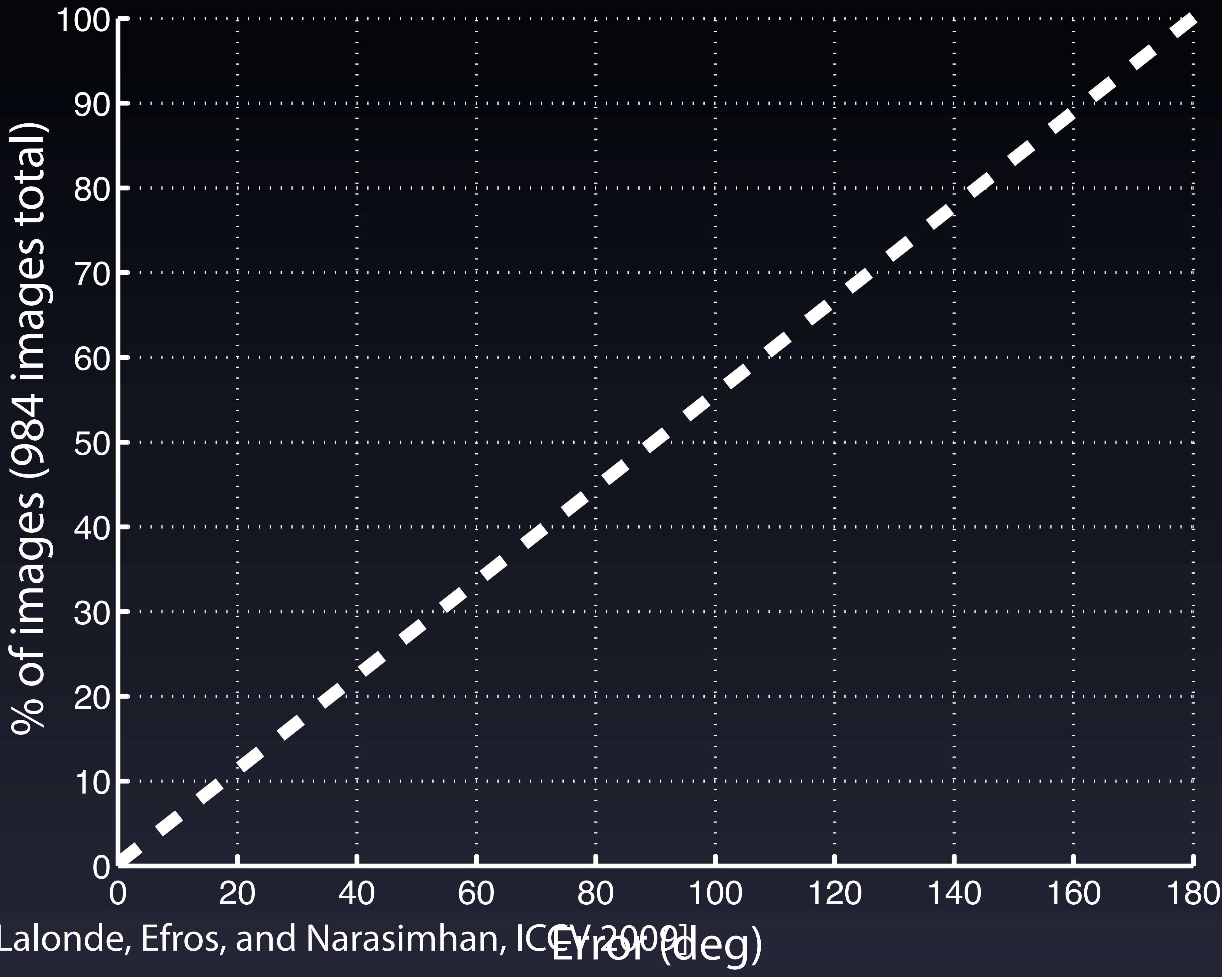


[Lalonde, Efros, and Narasimhan, ICCV 2009]

Here we show the cumulative plot of the % of images that have error less than the x axis.
This is what chance would look like.
Our scene cues, ...

Let me highlight two points here.
We're able to predict the sun position within 22 degrees for 55% of the images. This is equivalent to distinguishing between north-east and north-north-east, for instance. And for 75% of the images, we can distinguish between north and north-east for example.

Quantitative evaluation

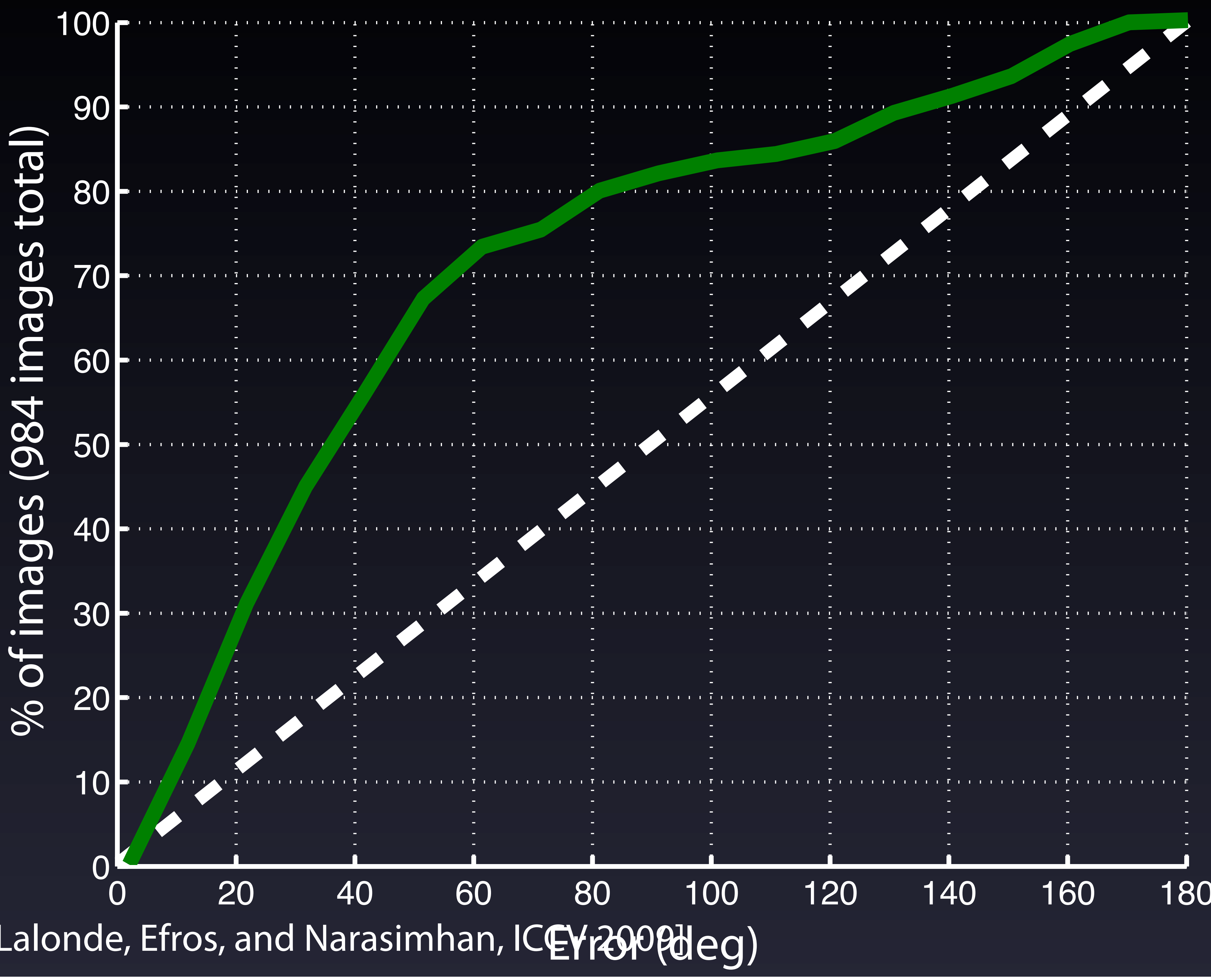


[Lalonde, Efros, and Narasimhan, ICCV 2009]

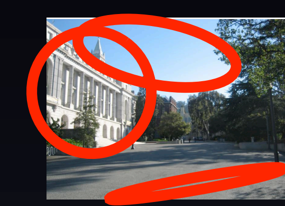
Here we show the cumulative plot of the % of images that have error less than the x axis.
This is what chance would look like.
Our scene cues, ...

Let me highlight two points here.
We’re able to predict the sun position within 22 degrees for 55% of the images. This is equivalent to distinguishing between north-east and north-north-east, for instance. And for 75% of the images, we can distinguish between north and north-east for example.

Quantitative evaluation



3 scene cues

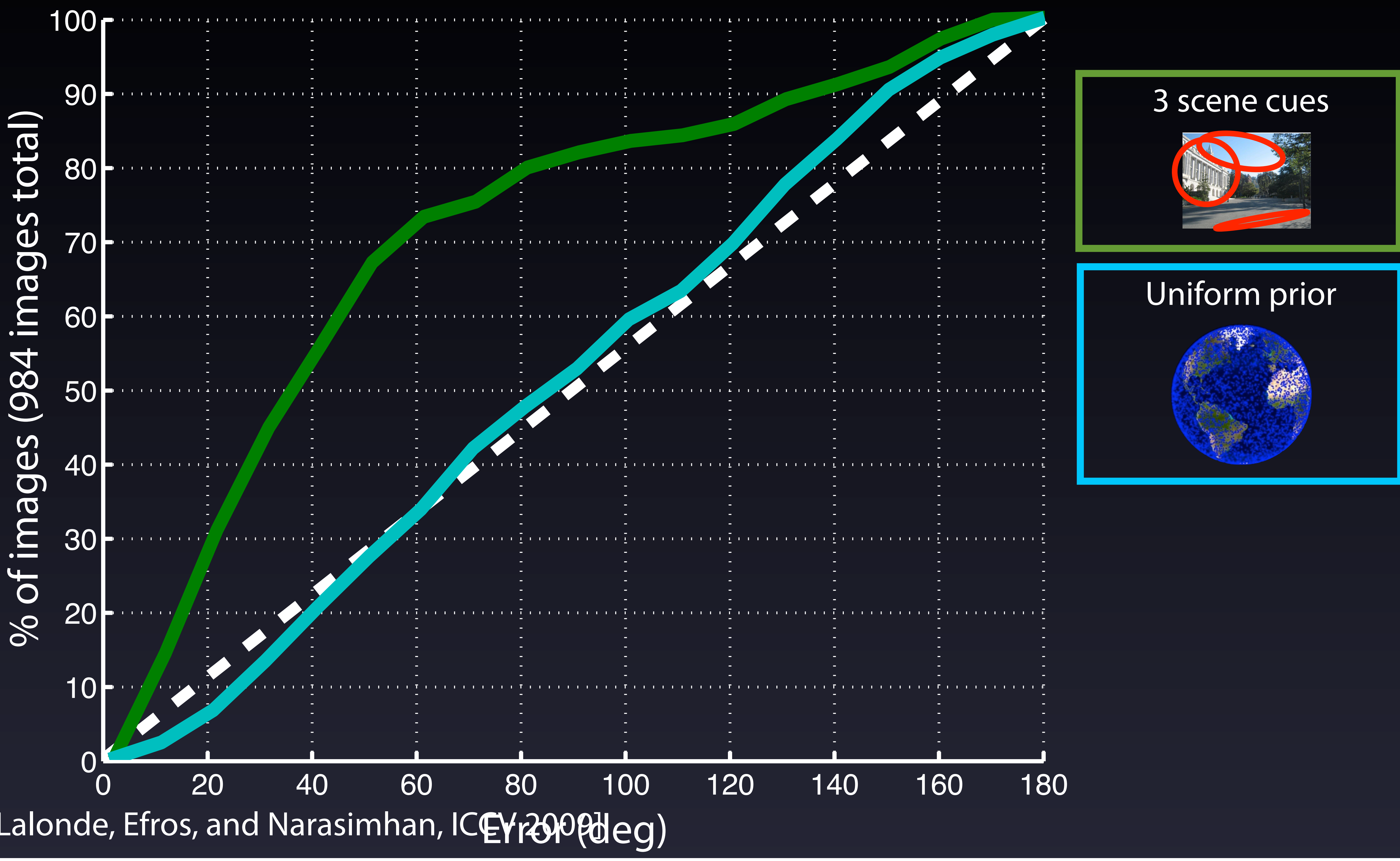


[Lalonde, Efros, and Narasimhan, ICCV 2009]

Here we show the cumulative plot of the % of images that have error less than the x axis.
This is what chance would look like.
Our scene cues, ...

Let me highlight two points here.
We're able to predict the sun position within 22 degrees for 55% of the images. This is equivalent to distinguishing between north-east and north-north-east, for instance. And for 75% of the images, we can distinguish between north and north-east for example.

Quantitative evaluation

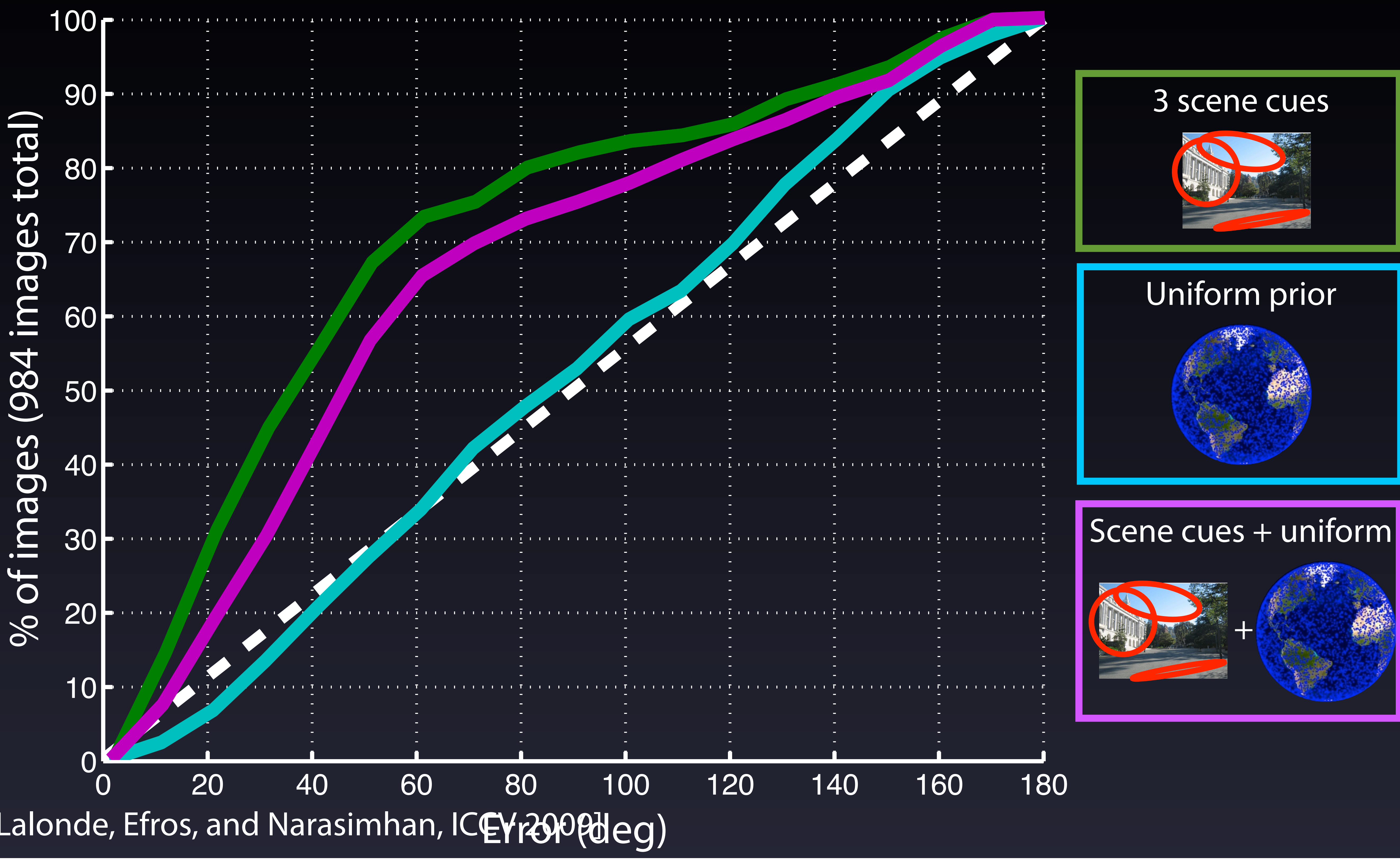


[Lalonde, Efros, and Narasimhan, ICCV 2009]

Here we show the cumulative plot of the % of images that have error less than the x axis.
This is what chance would look like.
Our scene cues, ...

Let me highlight two points here.
We're able to predict the sun position within 22 degrees for 55% of the images. This is equivalent to distinguishing between north-east and north-north-east, for instance. And for 75% of the images, we can distinguish between north and north-east for example.

Quantitative evaluation

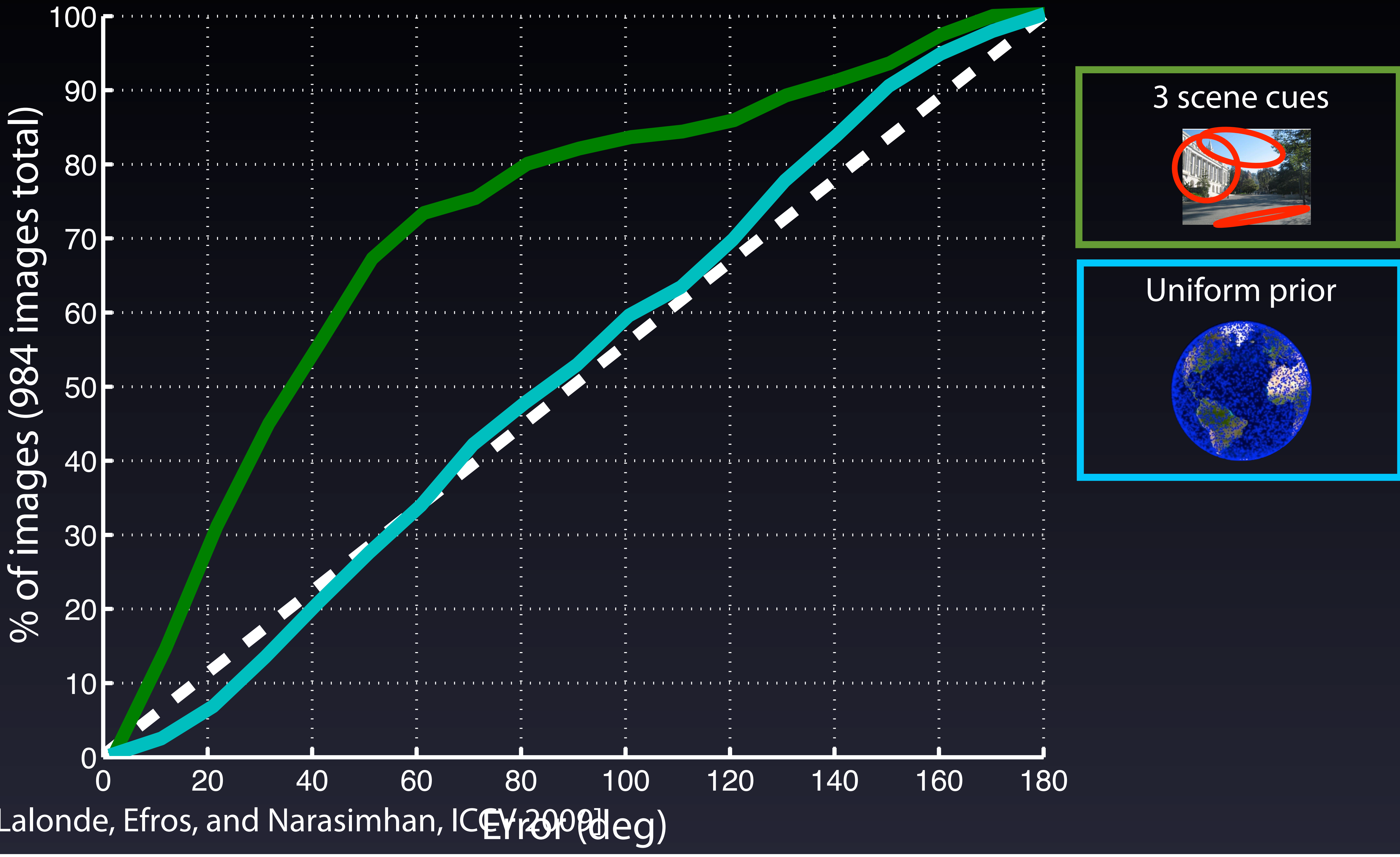


[Lalonde, Efros, and Narasimhan, ICCV 2009]

Here we show the cumulative plot of the % of images that have error less than the x axis.
This is what chance would look like.
Our scene cues, ...

Let me highlight two points here.
We’re able to predict the sun position within 22 degrees for 55% of the images. This is equivalent to distinguishing between north-east and north-north-east, for instance. And for 75% of the images, we can distinguish between north and north-east for example.

Quantitative evaluation

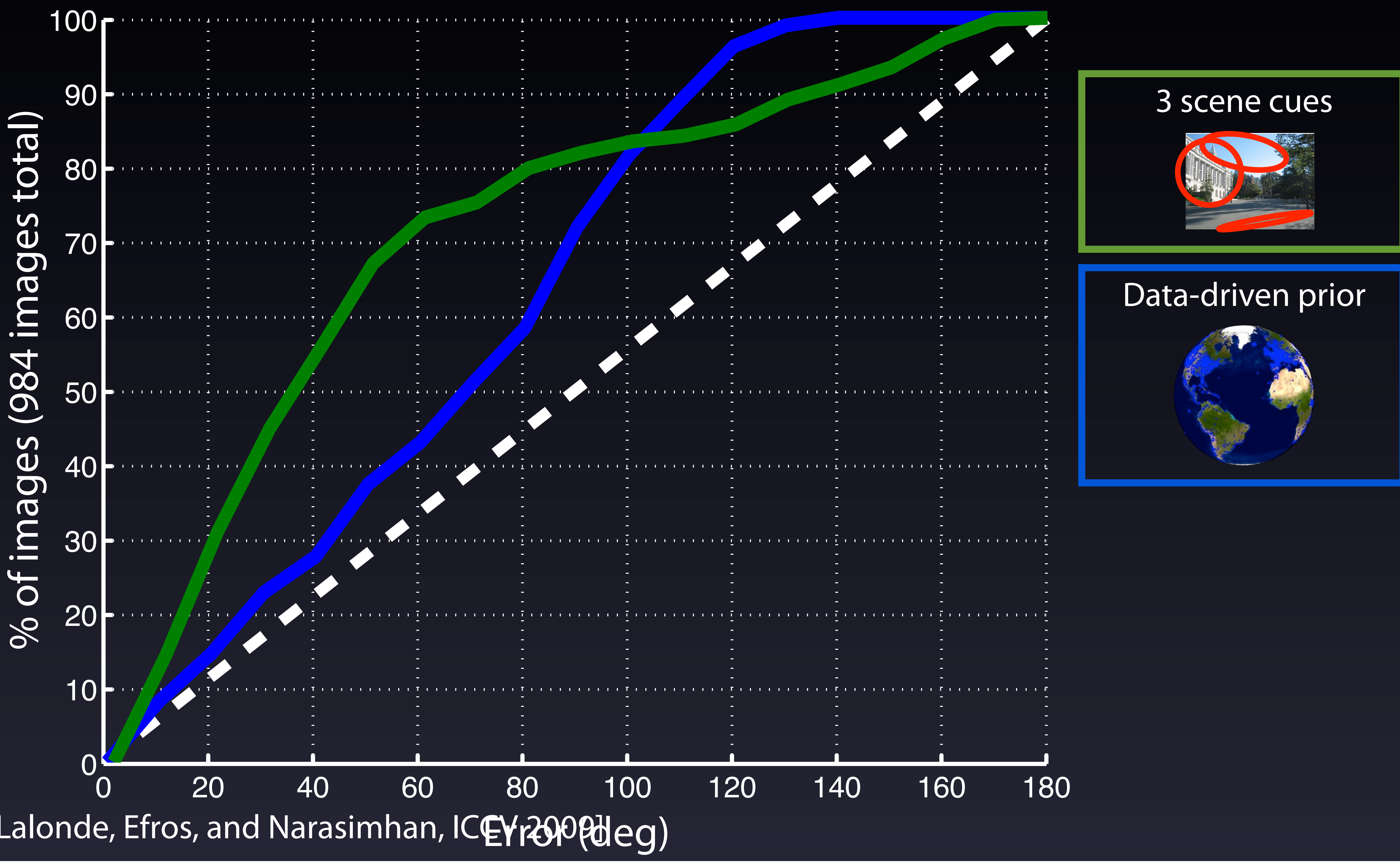


[Lalonde, Efros, and Narasimhan, ICCV 2009]

Here we show the cumulative plot of the % of images that have error less than the x axis.
This is what chance would look like.
Our scene cues, ...

Let me highlight two points here.
We're able to predict the sun position within 22 degrees for 55% of the images. This is equivalent to distinguishing between north-east and north-north-east, for instance. And for 75% of the images, we can distinguish between north and north-east for example.

Quantitative evaluation

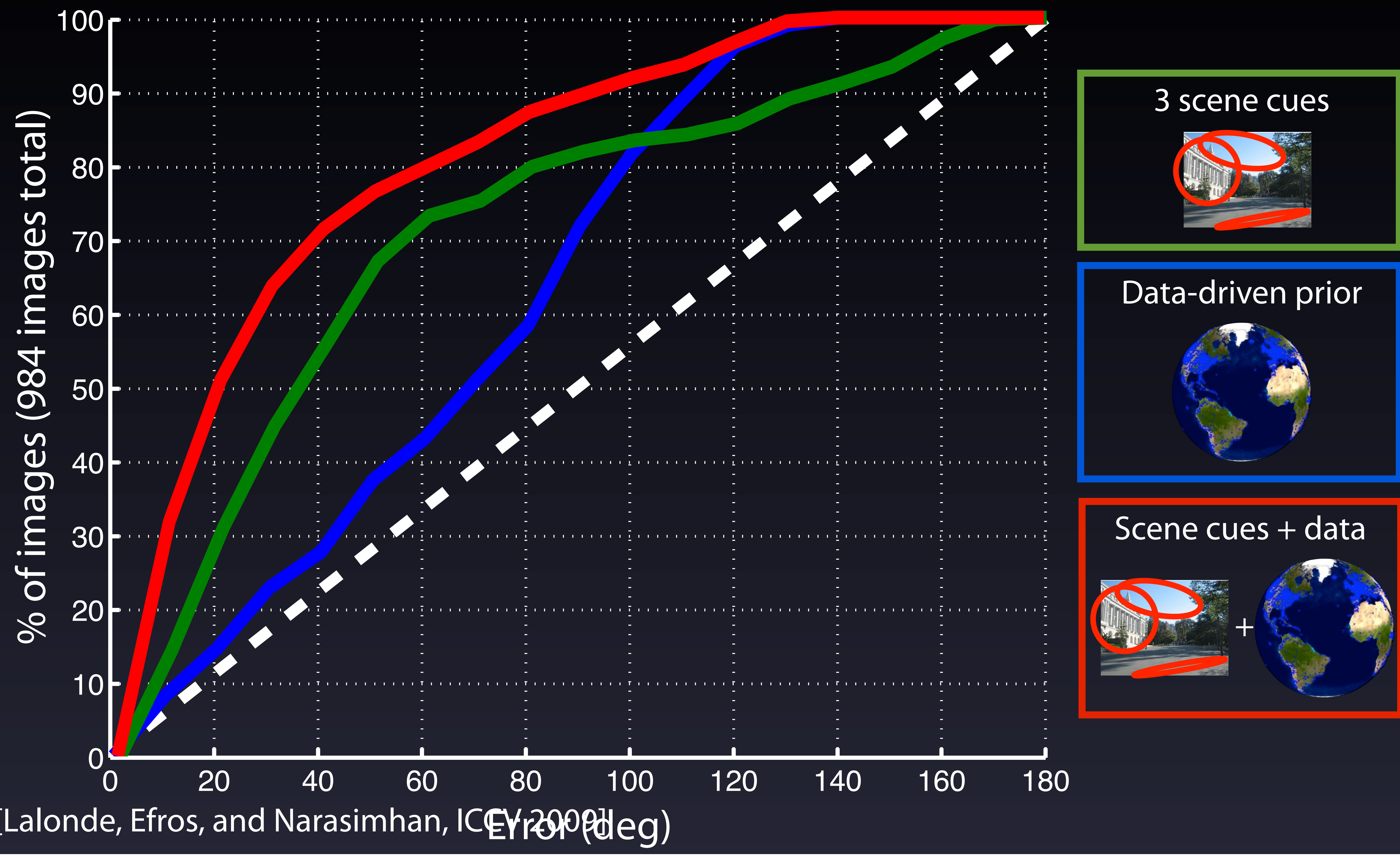


[Lalonde, Efros, and Narasimhan, ICCV 2009]

Here we show the cumulative plot of the % of images that have error less than the x axis.
This is what chance would look like.
Our scene cues, ...

Let me highlight two points here.
We're able to predict the sun position within 22 degrees for 55% of the images. This is equivalent to distinguishing between north-east and north-north-east, for instance. And for 75% of the images, we can distinguish between north and north-east for example.

Quantitative evaluation

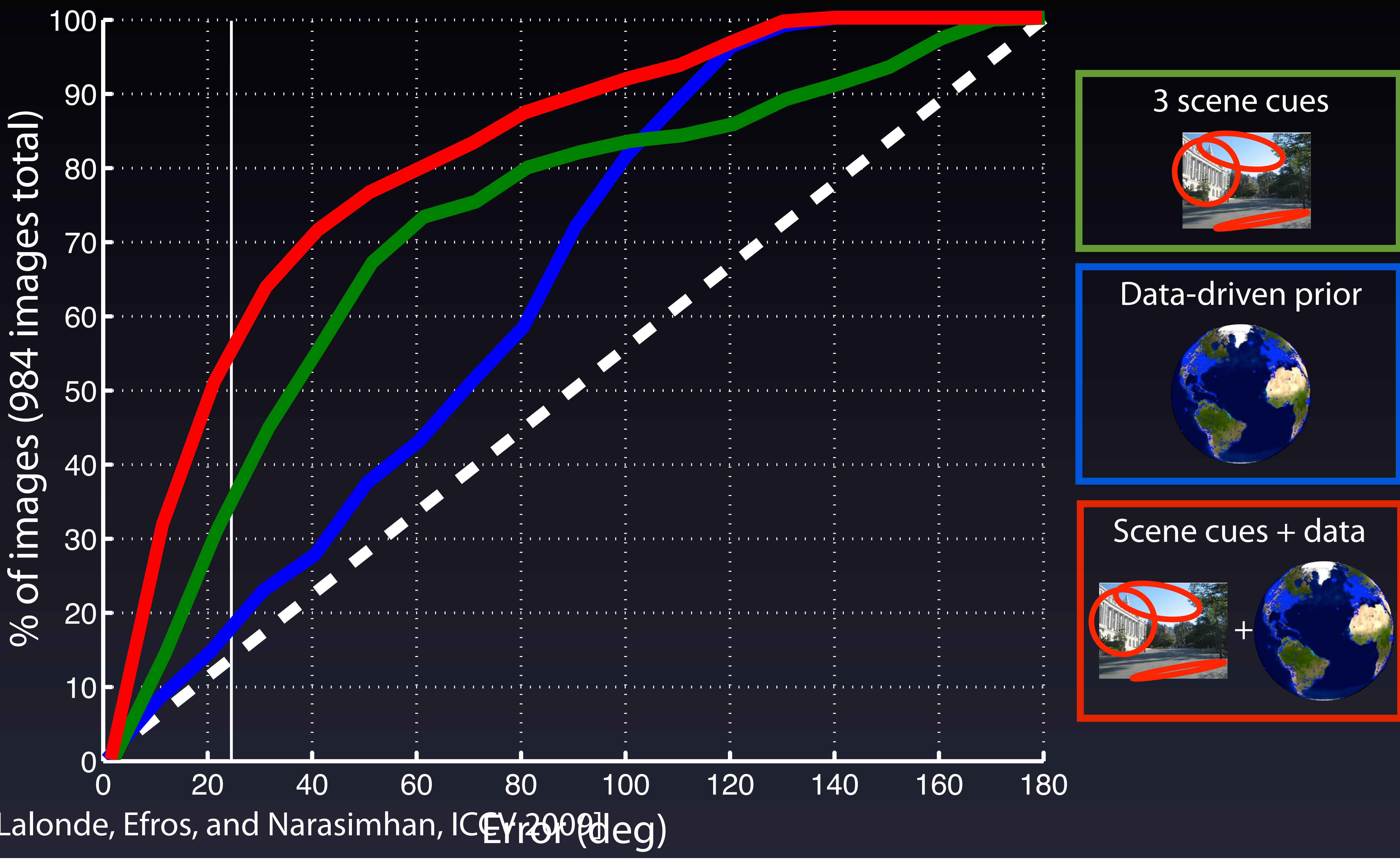


[Lalonde, Efros, and Narasimhan, ICCV 2009]

Here we show the cumulative plot of the % of images that have error less than the x axis.
This is what chance would look like.
Our scene cues, ...

Let me highlight two points here.
We're able to predict the sun position within 22 degrees for 55% of the images. This is equivalent to distinguishing between north-east and north-north-east, for instance. And for 75% of the images, we can distinguish between north and north-east for example.

Quantitative evaluation

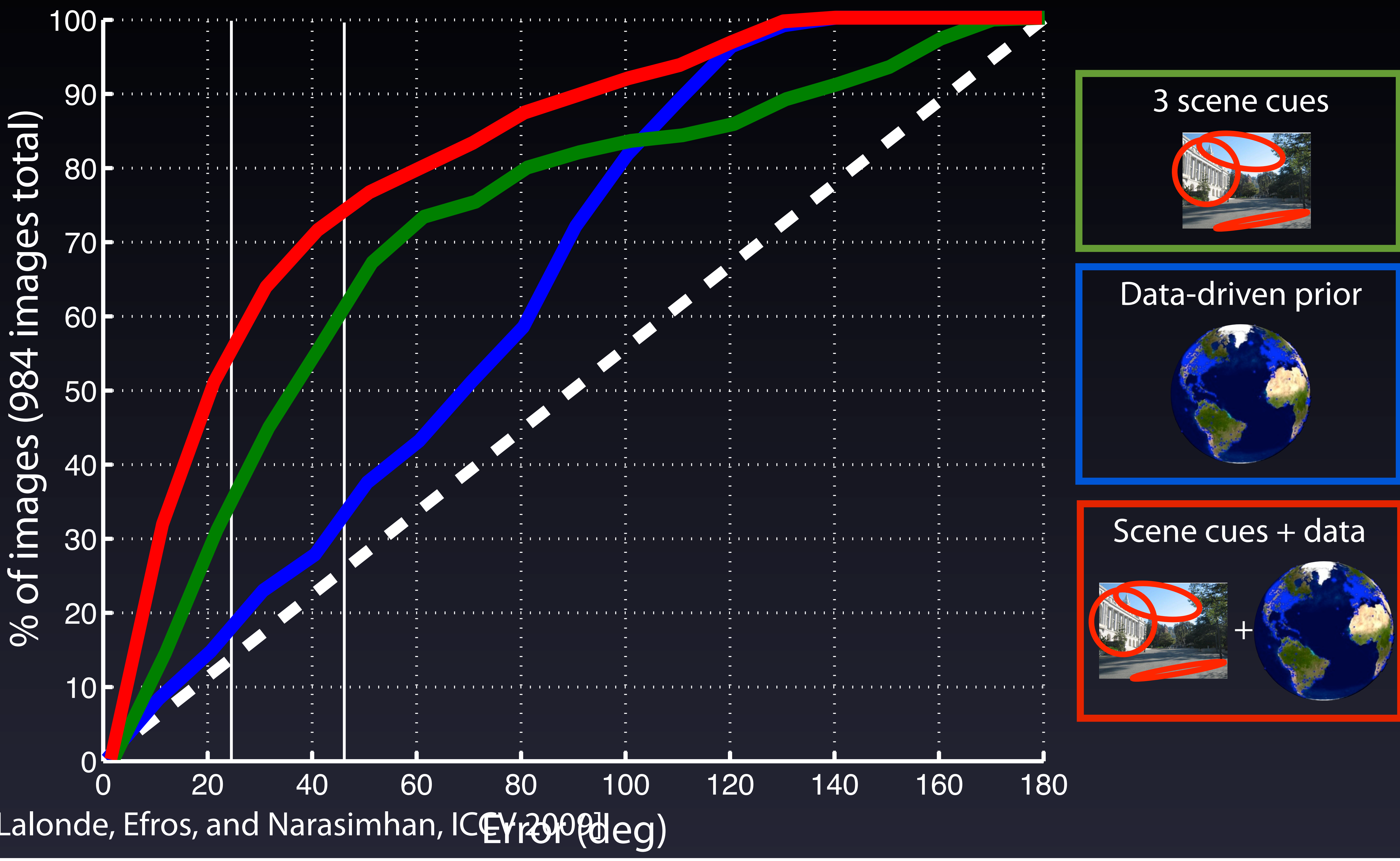


[Lalonde, Efros, and Narasimhan, ICCV 2009]

Here we show the cumulative plot of the % of images that have error less than the x axis.
This is what chance would look like.
Our scene cues, ...

Let me highlight two points here.
We're able to predict the sun position within 22 degrees for 55% of the images. This is equivalent to distinguishing between north-east and north-north-east, for instance. And for 75% of the images, we can distinguish between north and north-east for example.

Quantitative evaluation



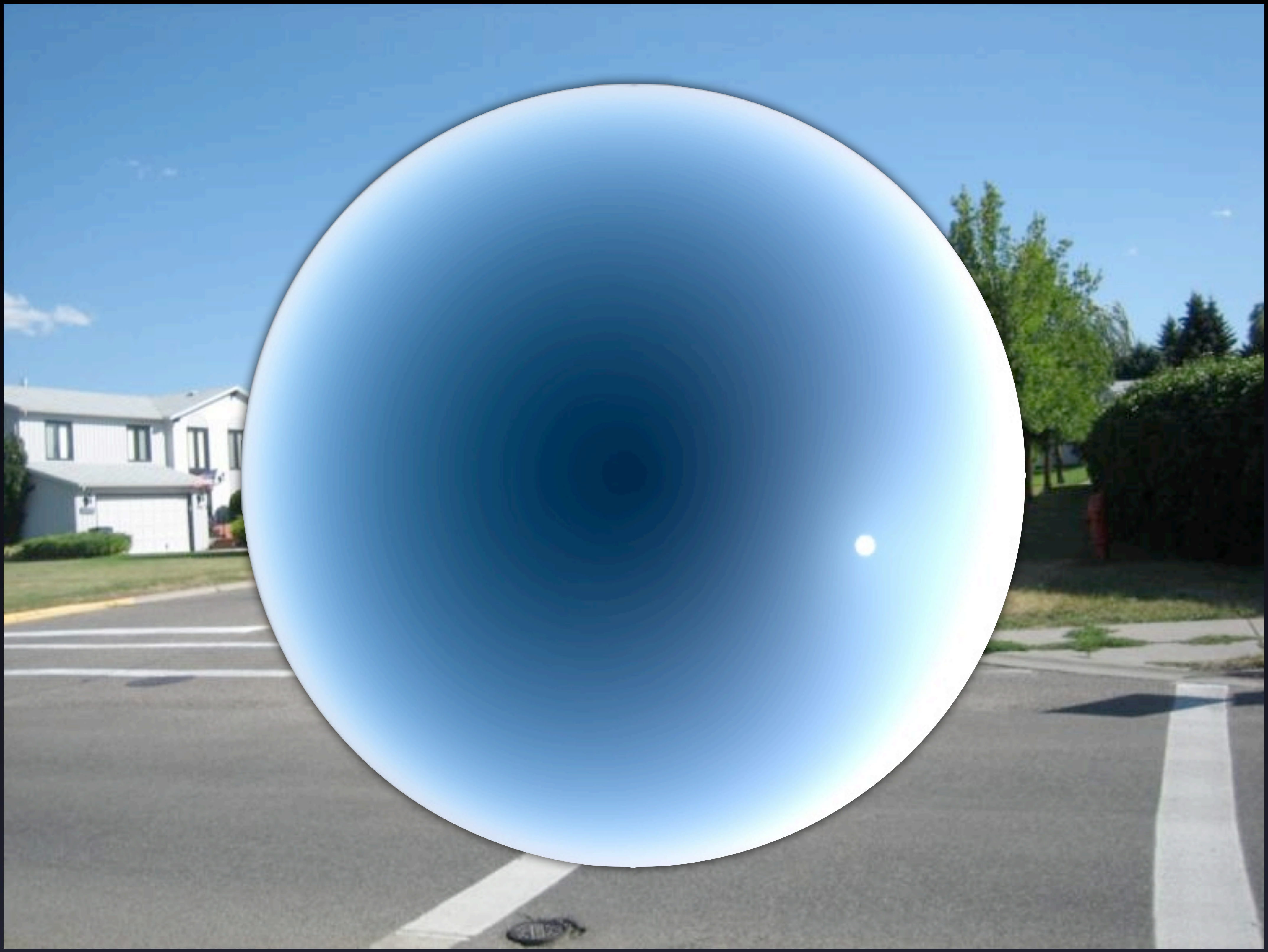
[Lalonde, Efros, and Narasimhan, ICCV 2009]

Here we show the cumulative plot of the % of images that have error less than the x axis.
This is what chance would look like.
Our scene cues, ...

Let me highlight two points here.
We're able to predict the sun position within 22 degrees for 55% of the images. This is equivalent to distinguishing between north-east and north-north-east, for instance. And for 75% of the images, we can distinguish between north and north-east for example.

[Lalonde, Efros, and Narasimhan, ICCV 2009]

One application of this method is object insertion.
When the sky is visible in the image, and when it is clear, we can actually fit our sky model to the most likely sun position, and synthesize the entire sky hemisphere!



[Lalonde, Efros, and Narasimhan, ICCV 2009]

One application of this method is object insertion.
When the sky is visible in the image, and when it is clear, we can actually fit our sky model to the most likely sun position, and synthesize the entire sky hemisphere!

[Lalonde, Efros, and Narasimhan, ICCV 2009]

We can use this high-dynamic range sky probe to render an object and realistically insert it in the image, and do so completely automatically.

[Lalonde, Efros, and Narasimhan, ICCV 2009]

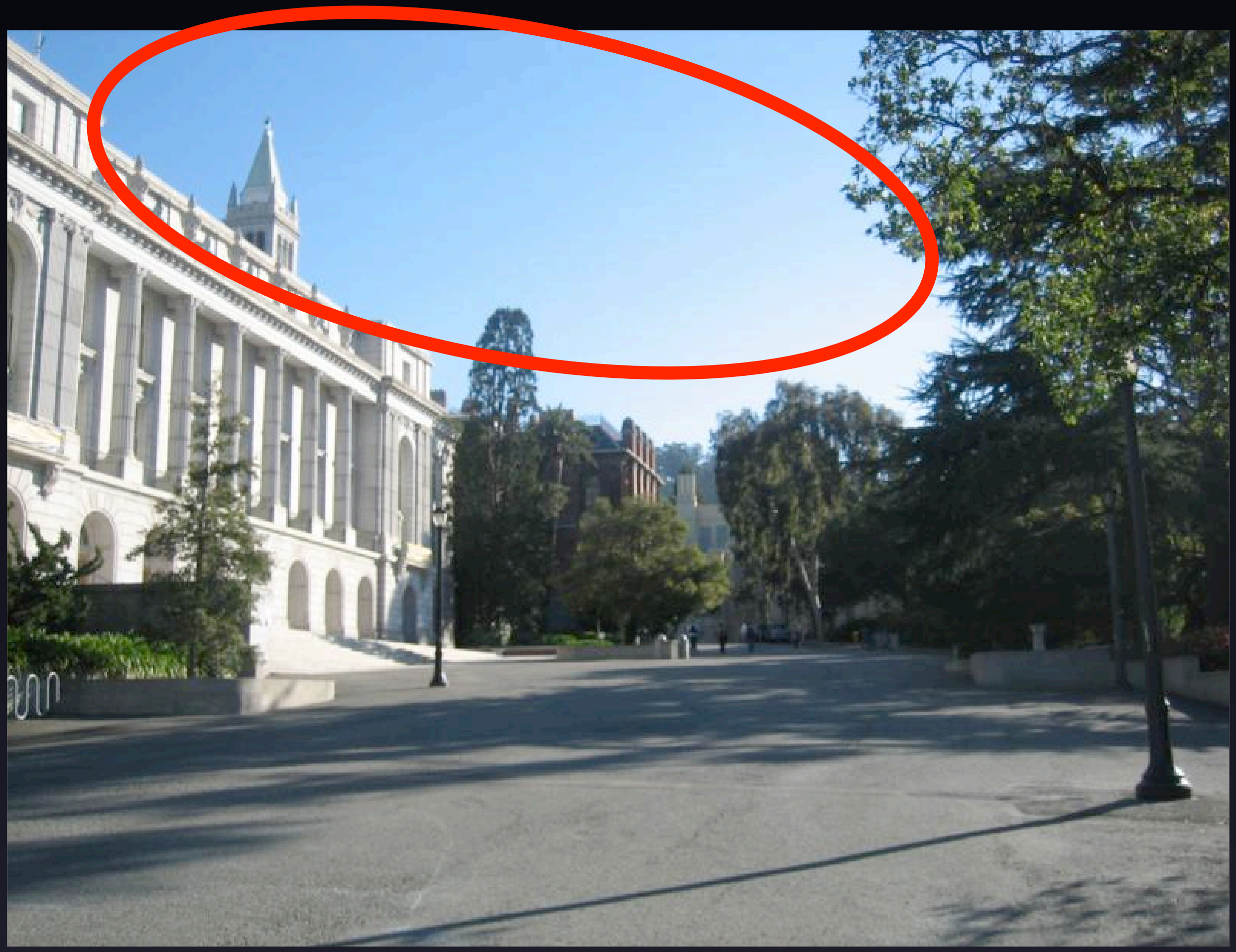
We can use this high-dynamic range sky probe to render an object and realistically insert it in the image, and do so completely automatically.

Conclusion

[Lalonde, Efros, and Narasimhan, ICCV 2009]

To conclude, we looked at 3 cues in the image which contain information about the sun relative position: the sky, shadows cast on the ground, and the shading on vertical surfaces. We showed that we can reliably estimate the relative sun position by combining the predictions of these cues together with a data-driven prior computed on 6M images.
These ideas have allowed us for the first time, to obtain information about illumination on uncontrolled, single outdoor images.
In our future work, we want to use that knowledge about illumination to introduce this idea of illumination-aware scene interpretation: can our estimate of illumination, even if it’s uncertain, to improve tasks like object detection and scene understanding.

Conclusion



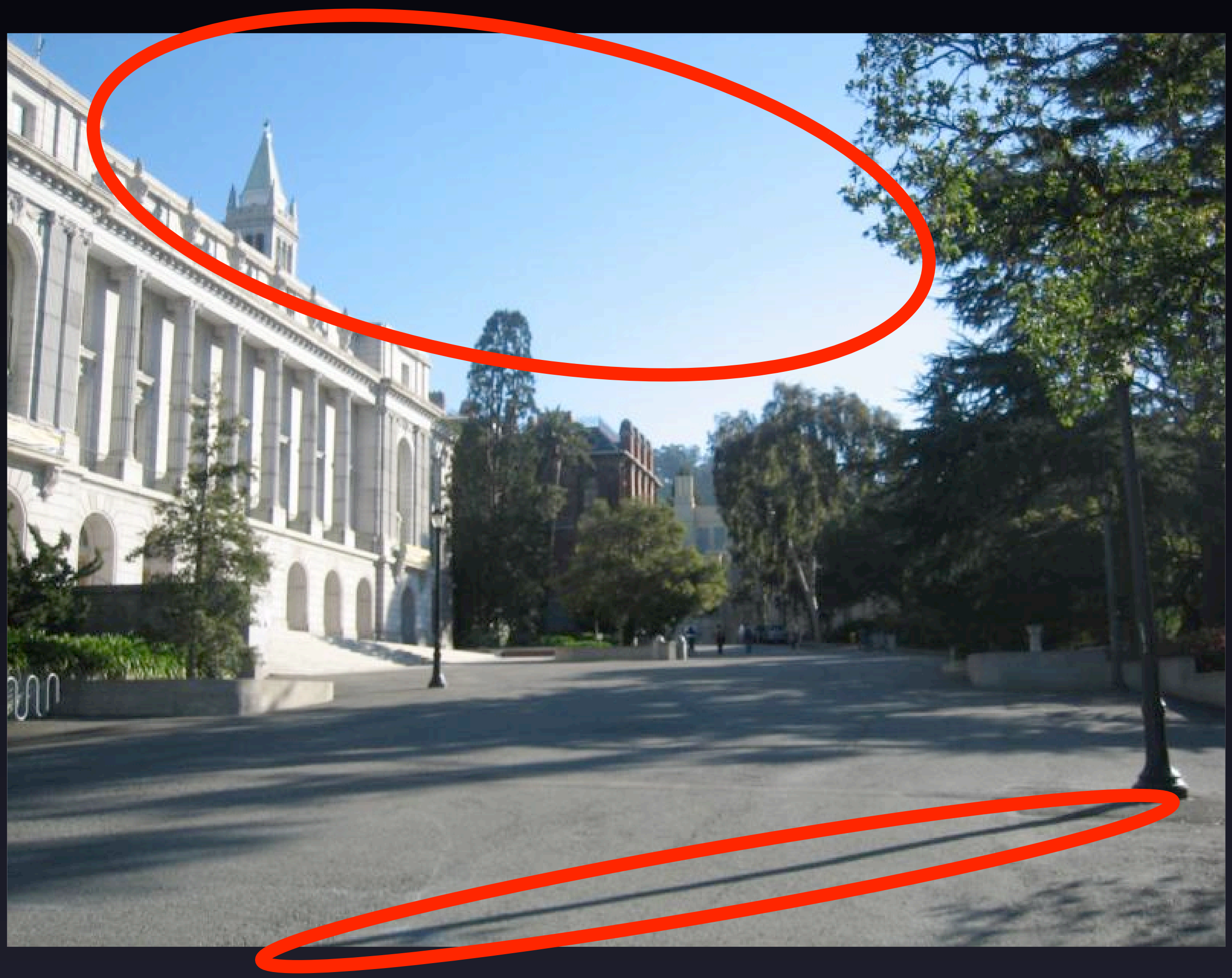
[Lalonde, Efros, and Narasimhan, ICCV 2009]

To conclude, we looked at 3 cues in the image which contain information about the sun relative position: the sky, shadows cast on the ground, and the shading on vertical surfaces. We showed that we can reliably estimate the relative sun position by combining the predictions of these cues together with a data-driven prior computed on 6M images.

These ideas have allowed us for the first time, to obtain information about illumination on uncontrolled, single outdoor images.

In our future work, we want to use that knowledge about illumination to introduce this idea of illumination-aware scene interpretation: can our estimate of illumination, even if it's uncertain, to improve tasks like object detection and scene understanding.

Conclusion

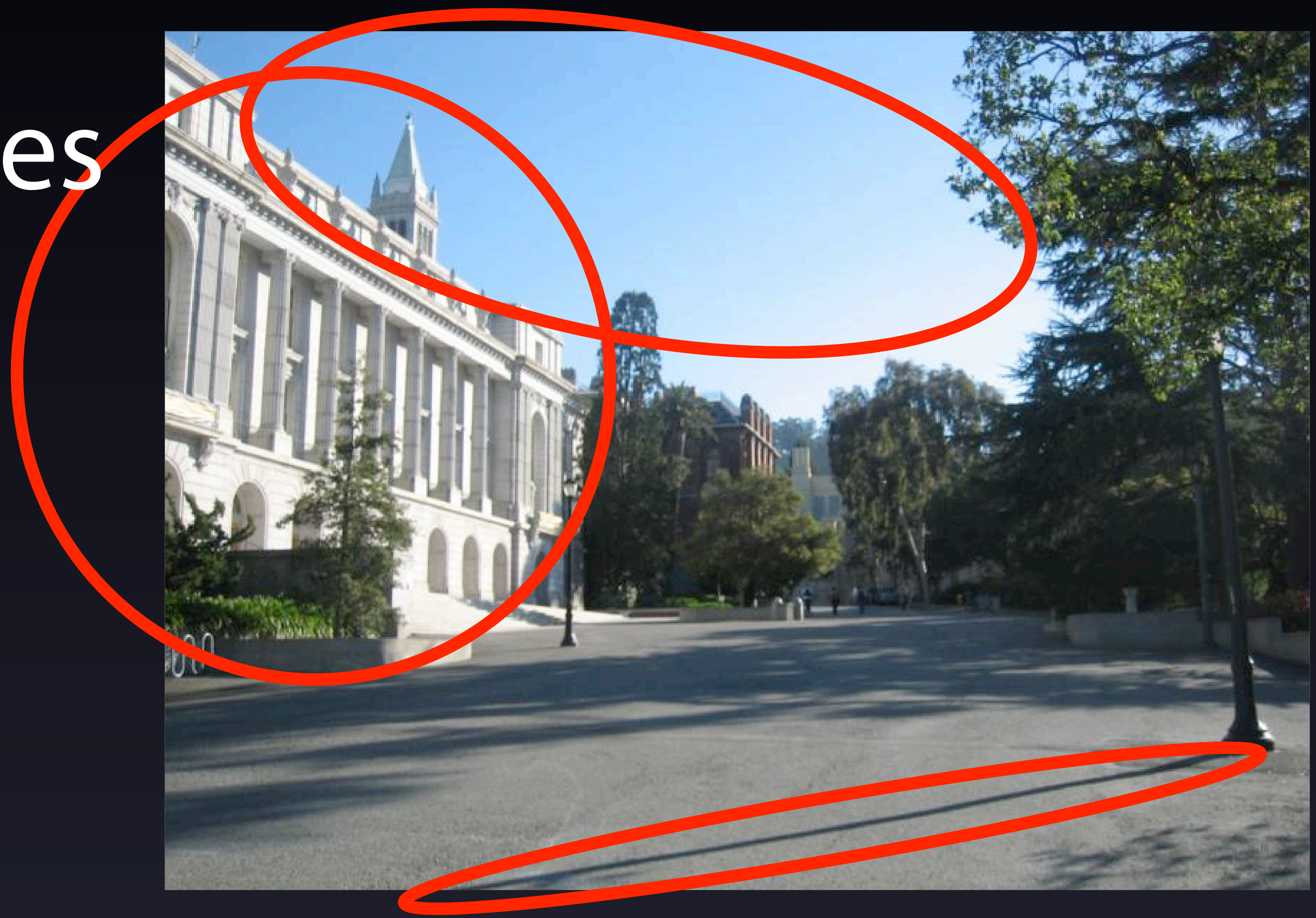


[Lalonde, Efros, and Narasimhan, ICCV 2009]

To conclude, we looked at 3 cues in the image which contain information about the sun relative position: the sky, shadows cast on the ground, and the shading on vertical surfaces. We showed that we can reliably estimate the relative sun position by combining the predictions of these cues together with a data-driven prior computed on 6M images.
These ideas have allowed us for the first time, to obtain information about illumination on uncontrolled, single outdoor images.
In our future work, we want to use that knowledge about illumination to introduce this idea of illumination-aware scene interpretation: can our estimate of illumination, even if it's uncertain, to improve tasks like object detection and scene understanding.

Conclusion

- sky, shadows, surfaces

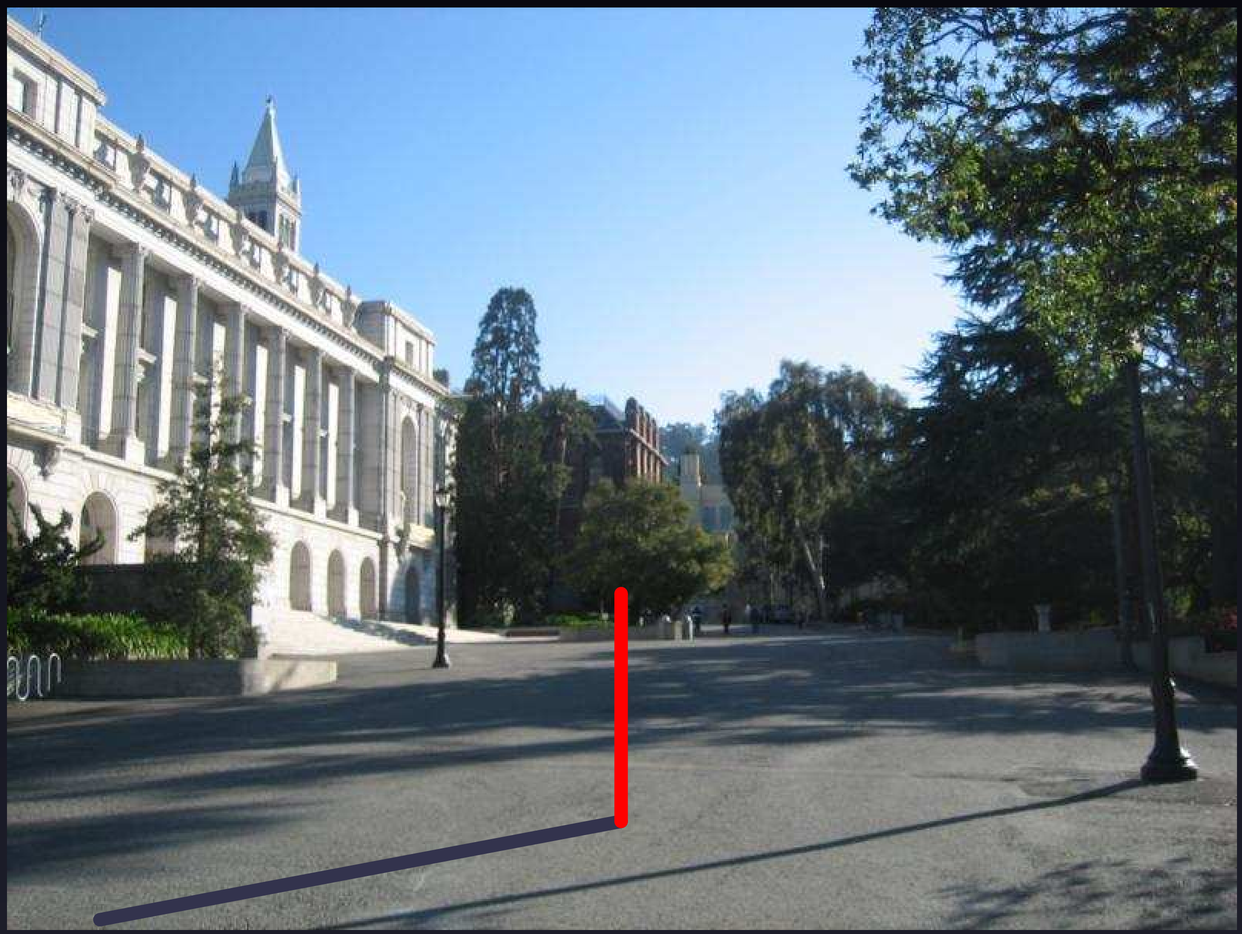


[Lalonde, Efros, and Narasimhan, ICCV 2009]

To conclude, we looked at 3 cues in the image which contain information about the sun relative position: the sky, shadows cast on the ground, and the shading on vertical surfaces. We showed that we can reliably estimate the relative sun position by combining the predictions of these cues together with a data-driven prior computed on 6M images.
These ideas have allowed us for the first time, to obtain information about illumination on uncontrolled, single outdoor images.
In our future work, we want to use that knowledge about illumination to introduce this idea of illumination-aware scene interpretation: can our estimate of illumination, even if it's uncertain, to improve tasks like object detection and scene understanding.

Conclusion

- sky, shadows, surfaces

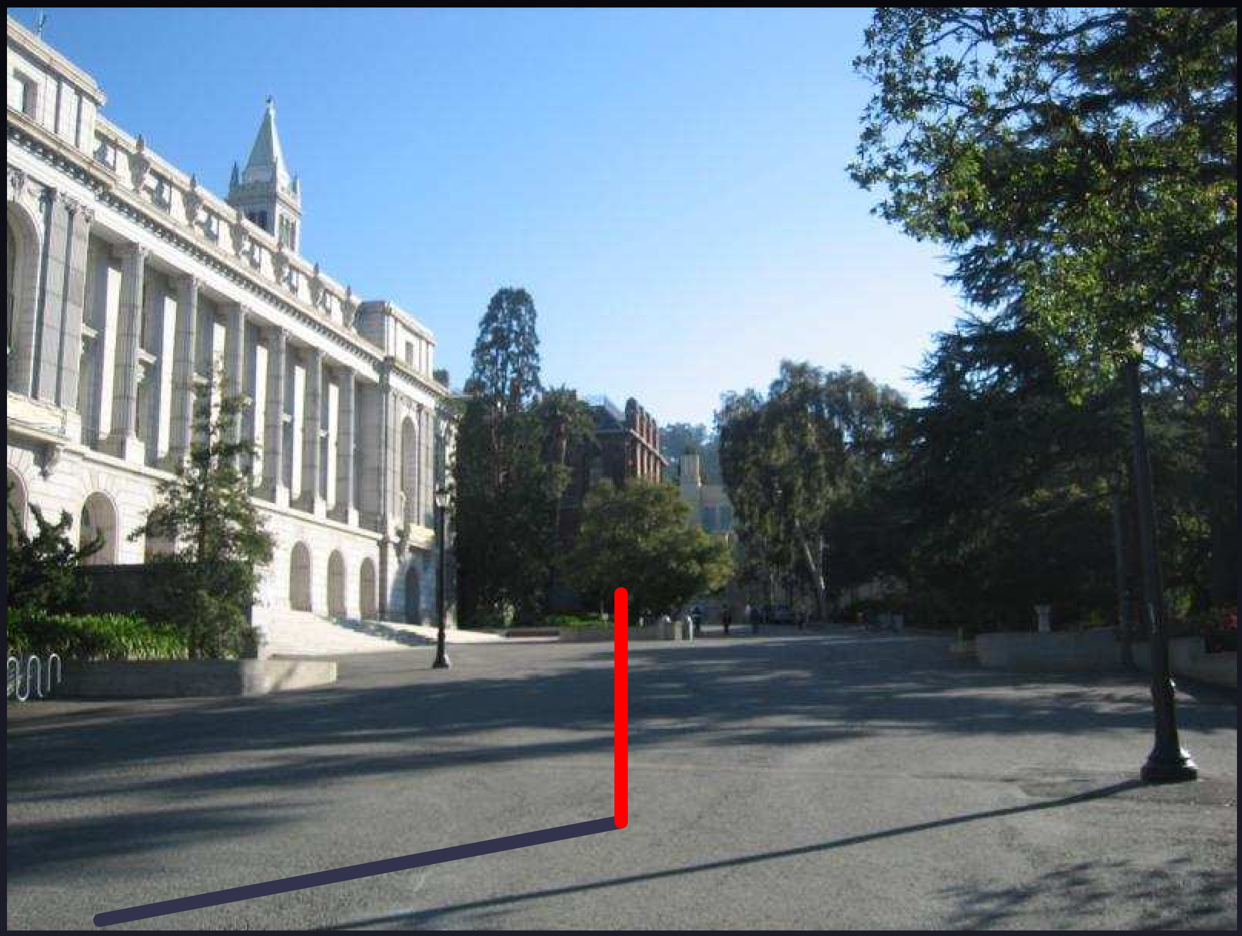


[Lalonde, Efros, and Narasimhan, ICCV 2009]

To conclude, we looked at 3 cues in the image which contain information about the sun relative position: the sky, shadows cast on the ground, and the shading on vertical surfaces. We showed that we can reliably estimate the relative sun position by combining the predictions of these cues together with a data-driven prior computed on 6M images.
These ideas have allowed us for the first time, to obtain information about illumination on uncontrolled, single outdoor images.
In our future work, we want to use that knowledge about illumination to introduce this idea of illumination-aware scene interpretation: can our estimate of illumination, even if it's uncertain, to improve tasks like object detection and scene understanding.

Conclusion

- sky, shadows, surfaces
- single, outdoor image

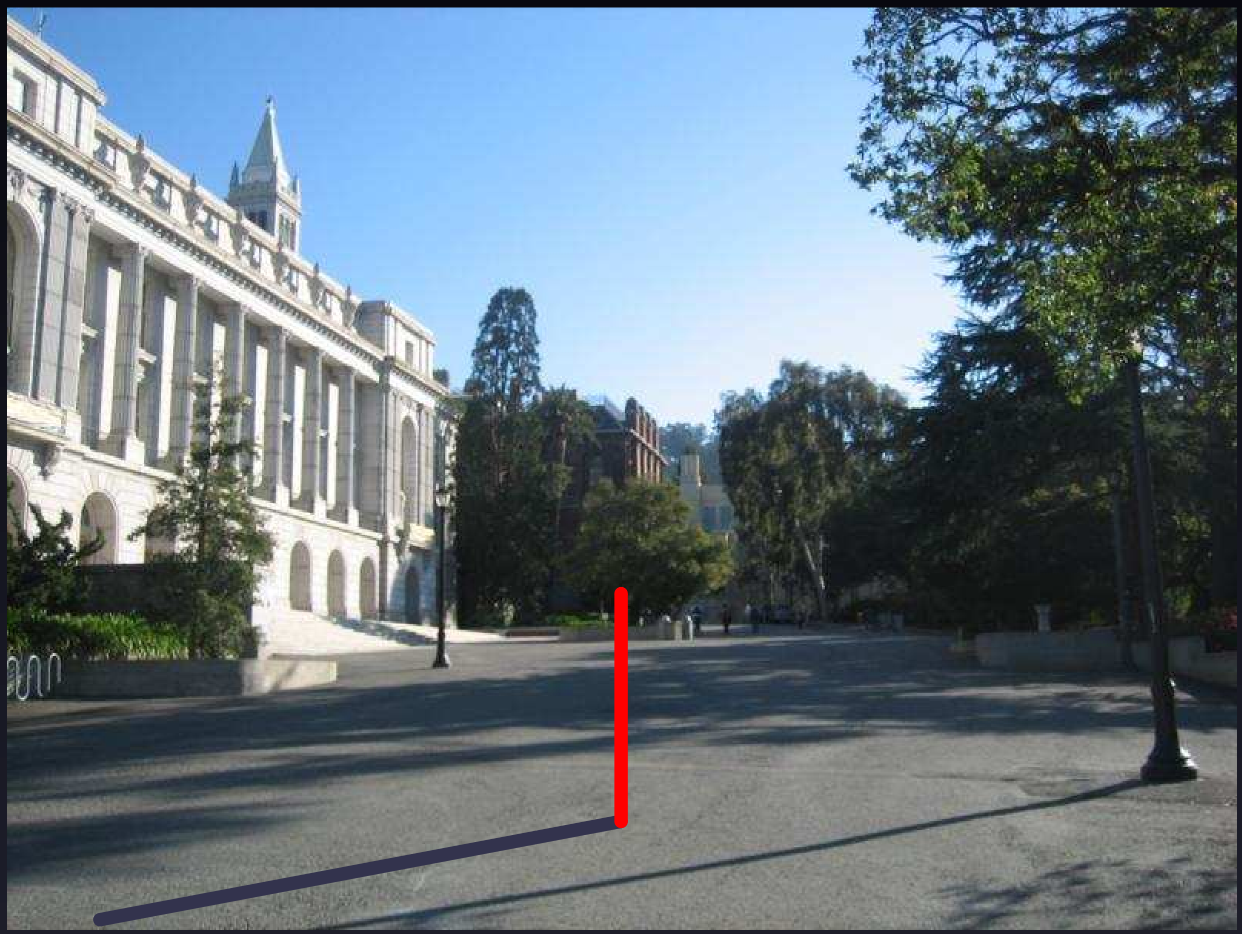


[Lalonde, Efros, and Narasimhan, ICCV 2009]

To conclude, we looked at 3 cues in the image which contain information about the sun relative position: the sky, shadows cast on the ground, and the shading on vertical surfaces. We showed that we can reliably estimate the relative sun position by combining the predictions of these cues together with a data-driven prior computed on 6M images.
These ideas have allowed us for the first time, to obtain information about illumination on uncontrolled, single outdoor images.
In our future work, we want to use that knowledge about illumination to introduce this idea of illumination-aware scene interpretation: can our estimate of illumination, even if it's uncertain, to improve tasks like object detection and scene understanding.

Conclusion

- sky, shadows, surfaces
- single, outdoor image
- “illumination aware” scene interpretation



[Lalonde, Efros, and Narasimhan, ICCV 2009]

To conclude, we looked at 3 cues in the image which contain information about the sun relative position: the sky, shadows cast on the ground, and the shading on vertical surfaces. We showed that we can reliably estimate the relative sun position by combining the predictions of these cues together with a data-driven prior computed on 6M images.
These ideas have allowed us for the first time, to obtain information about illumination on uncontrolled, single outdoor images.
In our future work, we want to use that knowledge about illumination to introduce this idea of illumination-aware scene interpretation: can our estimate of illumination, even if it’s uncertain, to improve tasks like object detection and scene understanding.