Inference 2 & Stereo Wrap-up

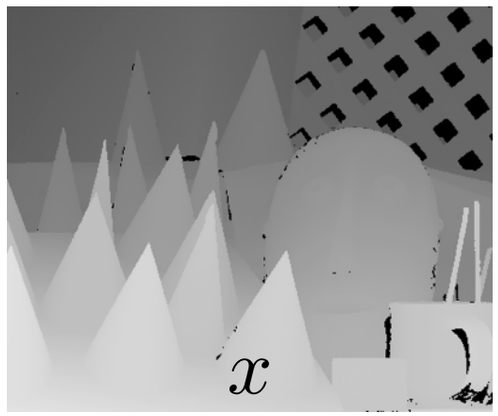
07.05.2014

TECHNISCHE UNIVERSITÄT DARMSTADT

visual inference

Reminder: Probabilistic Approach to Vision

Uncertain image measurements



Uncertain state of the world

- Model using posterior distribution: $p(\text{state}|\text{images})$
  - Describe the probability of the state of the world given the image measurements.
- How do we find the “best” state of the world?
  - Using probabilistic inference, e.g. we maximize w.r.t. state x

Reminder: Modeling the Posterior

- How can we model the posterior?
- We simplified the modeling problem by applying Bayes' rule (generative approach):

likelihood (observation model)

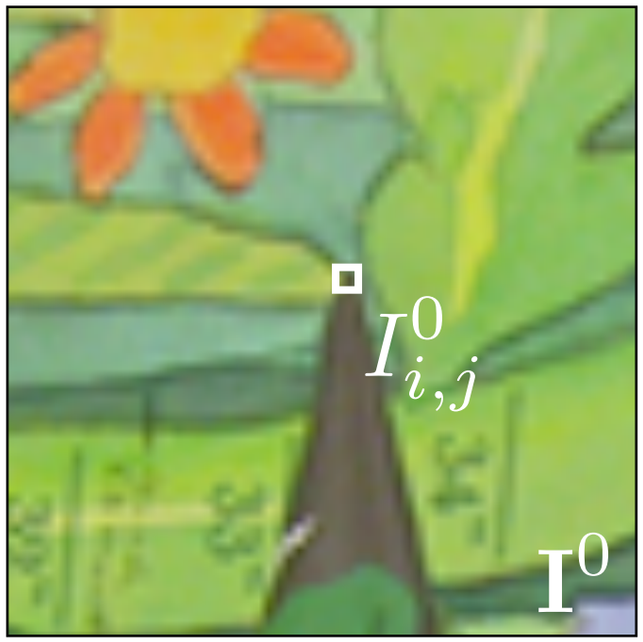
prior

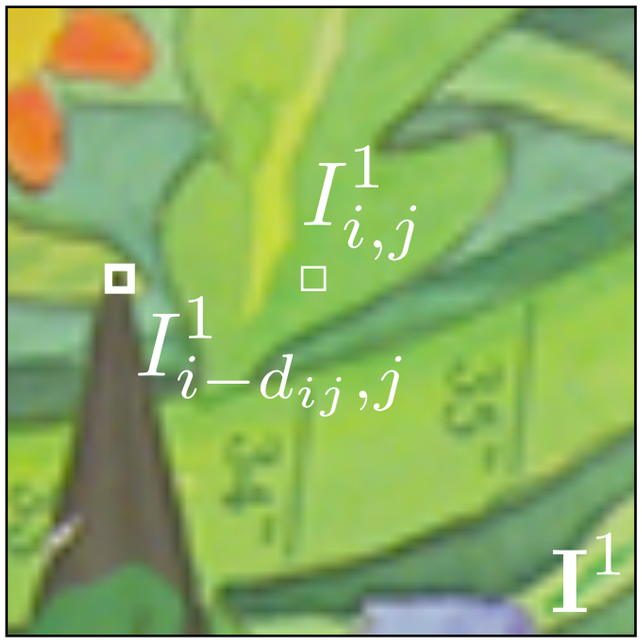
$$p(\text{state}|\text{images}) = \frac{p(\text{images}|\text{state}) \cdot p(\text{state})}{p(\text{images})}$$

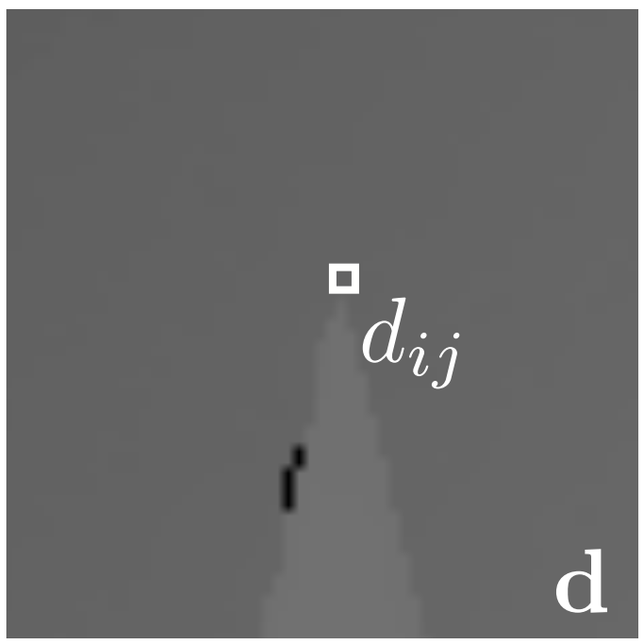
posterior

normalization term (constant)

Reminder Modeling the Likelihood







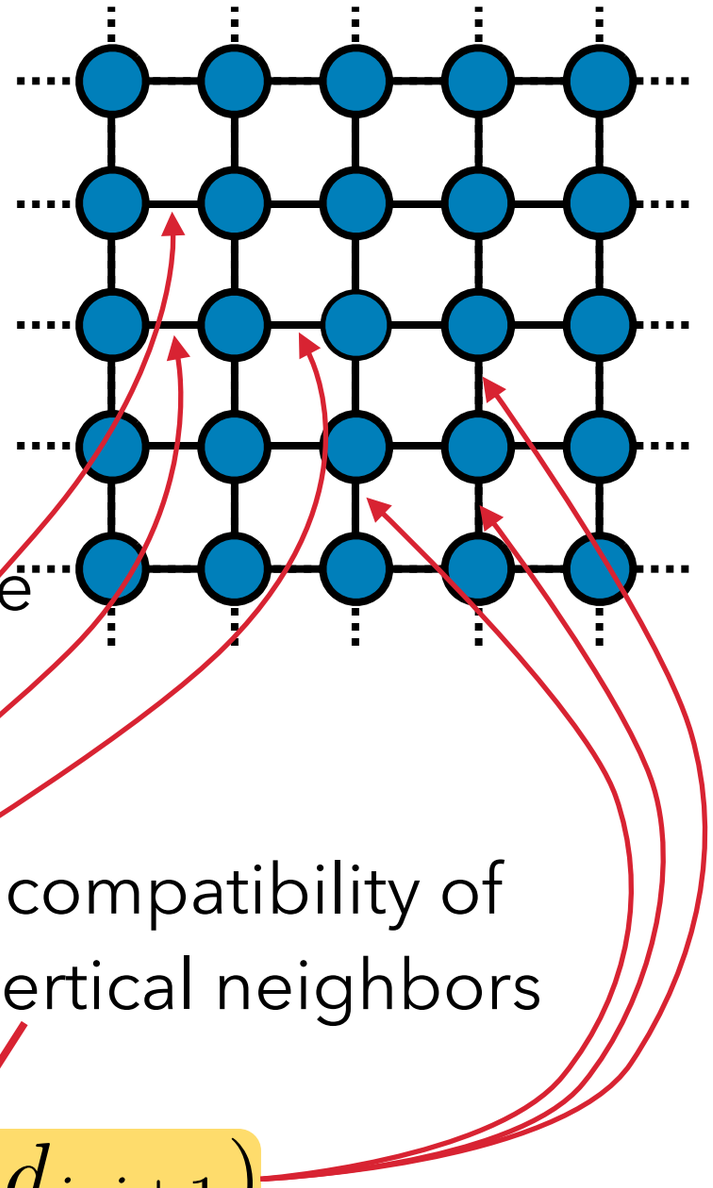
- A simple model:
  - We test how well the corresponding **pixels** match.

$$p(\mathbf{I}^0, \mathbf{I}^1 | \mathbf{d}) = \prod_{i,j} f(I^0_{i,j} - I^1_{(i-d_{ij}),j}) = \prod_{i,j} \mathcal{N}(I^0_{i,j} - I^1_{(i-d_{ij}),j};\ 0, \sigma^2)$$

Reminder: Markov Random Fields

TECHNISCHE UNIVERSITÄT DARMSTADT

- We formulated the prior as a so-called **graphical model**, or more specifically a so called **Markov random field**.
  - Each edge (in this particular graph) corresponds to a term in the prior that models how compatible two neighboring pixels are in terms of their disparity:



compatibility of horizontal neighbors

compatibility of vertical neighbors

$$p(\mathbf{d}) = \frac{1}{Z} \prod_{i,j} f_H(d_{i,j}, d_{i+1,j}) \cdot f_V(d_{i,j}, d_{i,j+1})$$

product over all the pixels

Potts Model

TECHNISCHE UNIVERSITÄT DARMSTADT

- Define very simple compatibility functions:

$$f_H(d_{i,j} - d_{i+1,j}) = \frac{1}{Z(T)} \exp\left\{ \frac{1}{T} \delta(d_{i,j} - d_{i+1,j}) \right\}$$

- Kronecker delta:

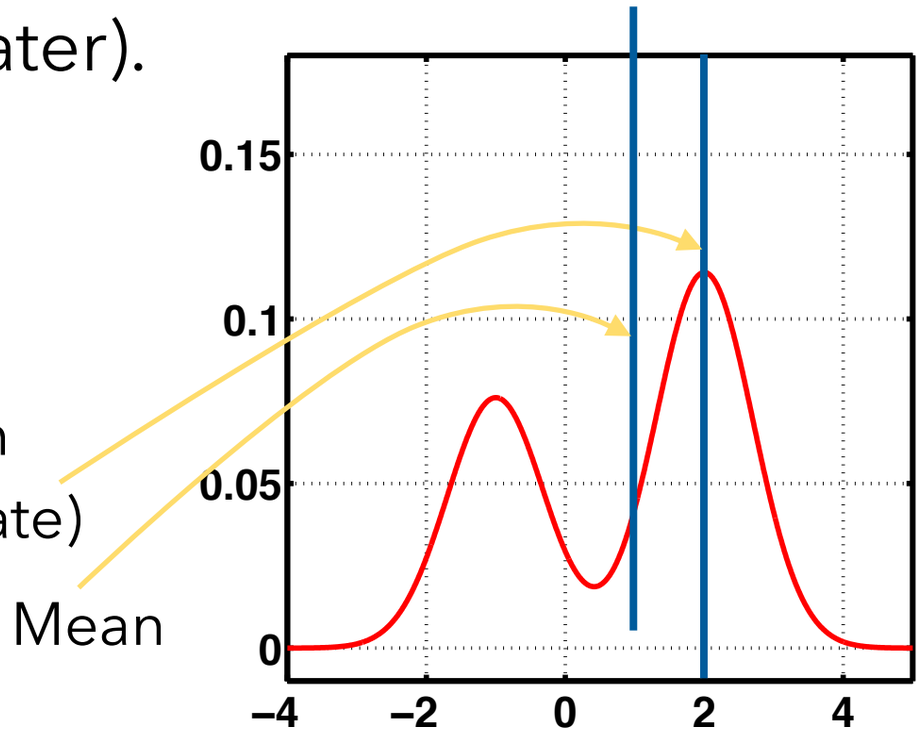
$$\delta(a-b) = \begin{cases} 1, & a = b \\ 0, & a \neq b \end{cases}$$

- This prior:
  - Prefers to have the same disparities at neighboring pixels.
  - But allows for disparity discontinuities with no penalty for large discontinuities.
  - Is called a **Potts model**.
    - Originally from statistical physics (magnetism)

Probabilistic Inference

TECHNISCHE UNIVERSITÄT DARMSTADT

- ... generally means one of three things:
  - Computing the maximum of the posterior distribution $p(x|y)$, that is computing the state that is the most probable given our observations (maximum a-posteriori (MAP) estimation).
  - Computing expectations over the posterior distribution, such as the mean of the posterior $p(x|y)$.
  - Computing marginal distributions (later).
- How do these differ?
  - Assume we have the following posterior distribution:
  - The posterior may be multi-modal!



Probabilistic Inference

TECHNISCHE UNIVERSITÄT DARMSTADT

- We will talk about MAP estimation first:
  - First "attempt": Continuous optimization methods.
  - Second "attempt": Graph-based methods (graph cuts).
- Later we will briefly get into computing expectations and marginal distributions:
  - Belief propagation or the sum-product algorithm.
    - This can be extended to MAP estimation: The max-product algorithm.

Continuous Optimization

TECHNISCHE UNIVERSITÄT DARMSTADT

- The most straightforward idea for maximizing the posterior is to apply well-known continuous optimization techniques.
- Especially gradient techniques have found widespread use, e.g.:
  - Simple **gradient ascent**, also called hill-climbing.
  - Conjugate gradient methods.
  - And many more.
- Since the posterior may be multi-modal (more on that later), this will typically only give us a **local optimum** and not the global optimum.

Discrete Optimization

TECHNISCHE UNIVERSITÄT DARMSTADT

- An alternative to using these continuous optimization techniques is to use discrete optimization.
  - For stereo this makes sense, because disparity is often described in increments of whole pixels: $d \in \{0, \ldots, D\}$
  - Big problem though: MAP estimation for general discrete MRFs is an **NP hard** problem [Shimony, 94].
    - We cannot really hope to get reasonably efficient algorithms for the general case.
- The good news: For important classes of MRFs, there are polynomial time algorithms.
  - The space optimized over is nevertheless exponentially large.
  - For an image of $N \times M$ pixels, we have $(D+1)^{N \times M}$ possible solutions.

Simplified Stereo Problem

TECHNISCHE UNIVERSITÄT DARMSTADT

- Let us assume that there are only two disparity levels: $d \in \{0, 1\}$
  - We label each disparity site as either on or off.
  - That is not realistic, but can be extended to the more general case.
  - Example:

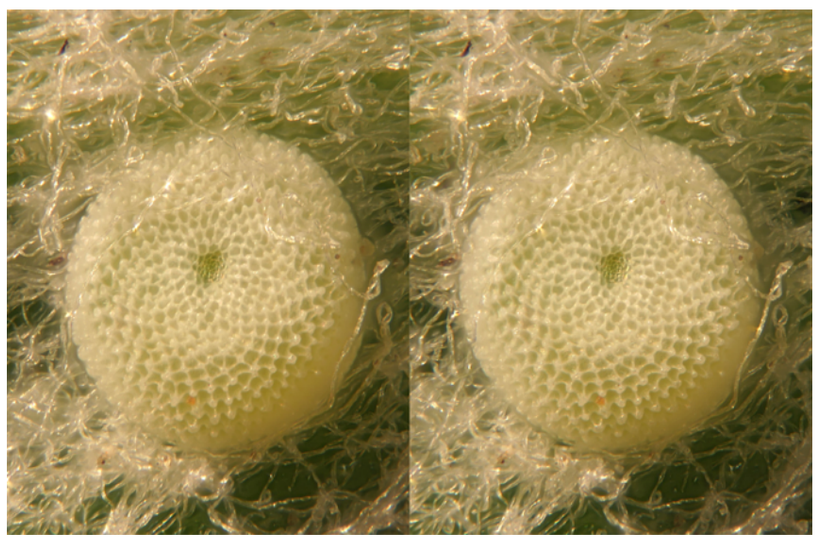
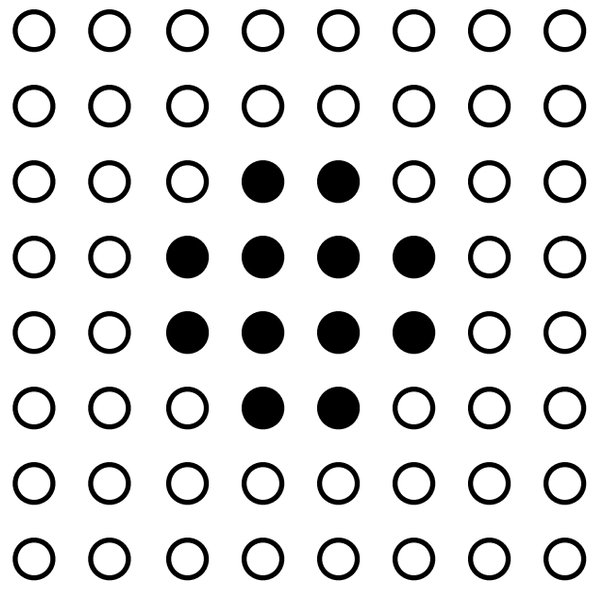


Image pair [Rik Littlefield]

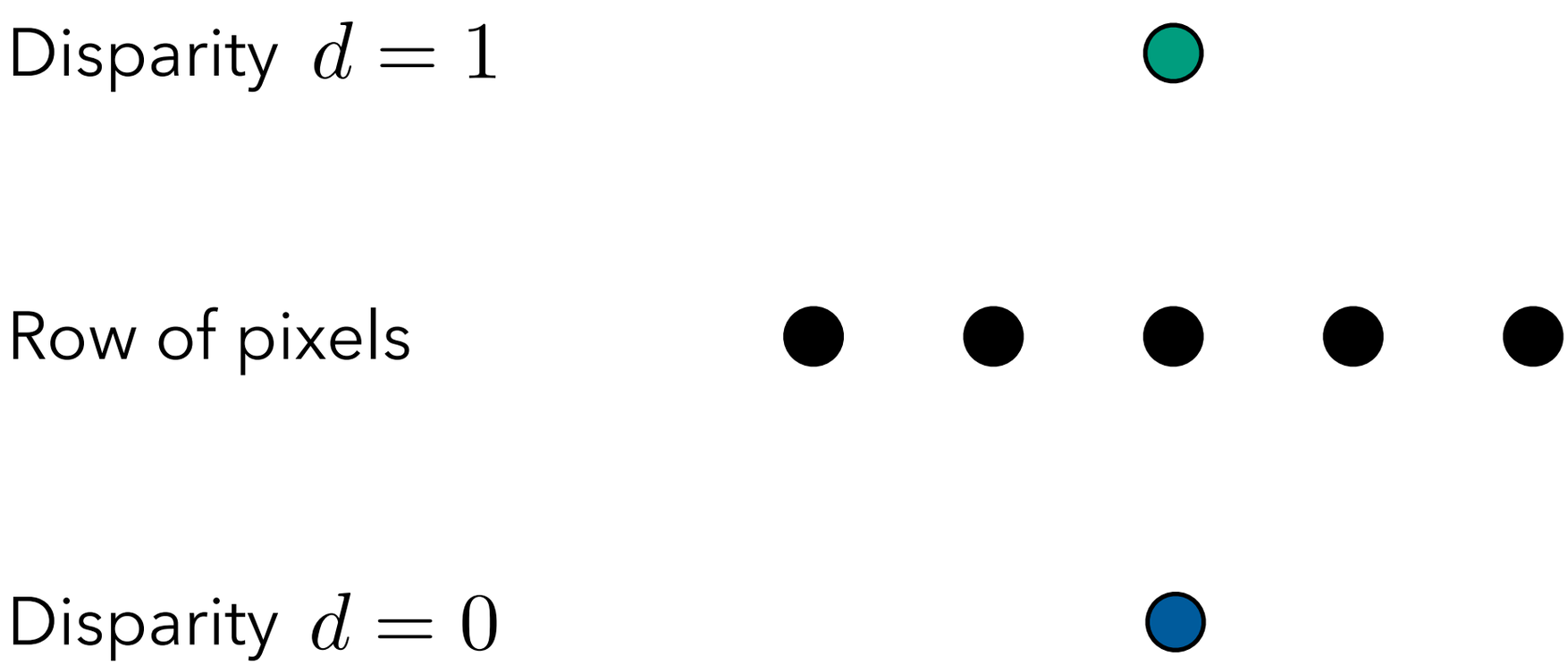


Disparity map (black - closer)

Graph Representation

TECHNISCHE UNIVERSITÄT DARMSTADT

- For now, let us only look at a single row of pixels.
- We can represent the energy (i.e. the negative log-posterior) as a graph:
  - Every pixel is a node (as before)
  - Two additional nodes, one for each disparity level



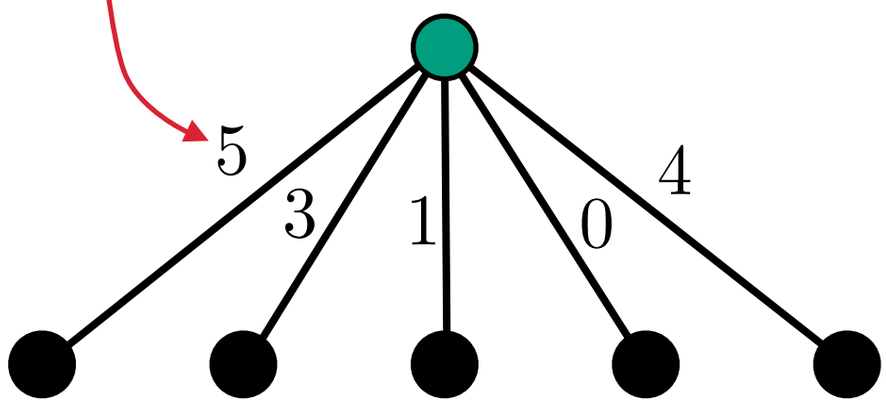
Graph Representation

TECHNISCHE UNIVERSITÄT DARMSTADT

- The edges of the graph indicate the cost of pairing two nodes in the solution:
  - We get them directly from our energy (negative log-posterior).

Cost for **not** assigning
disparity $d = 1$ to the pixel

Disparity $d = 1$



Row of pixels

Disparity $d = 0$

Graph Representation

TECHNISCHE UNIVERSITÄT DARMSTADT

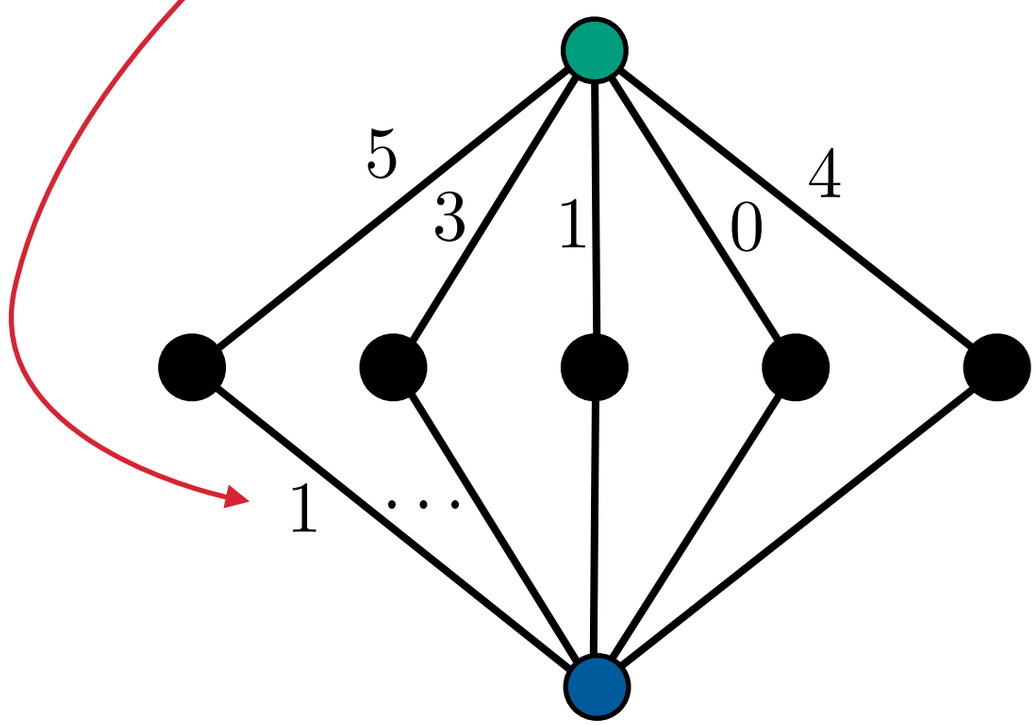
- The edges of the graph indicate the cost of pairing two nodes in the solution:
  - We get them directly from our energy (negative log-posterior).

Cost for **not** assigning disparity $d = 0$ to the pixel

Disparity $d = 1$

Row of pixels

Disparity $d = 0$



Graph Representation

TECHNISCHE UNIVERSITÄT DARMSTADT

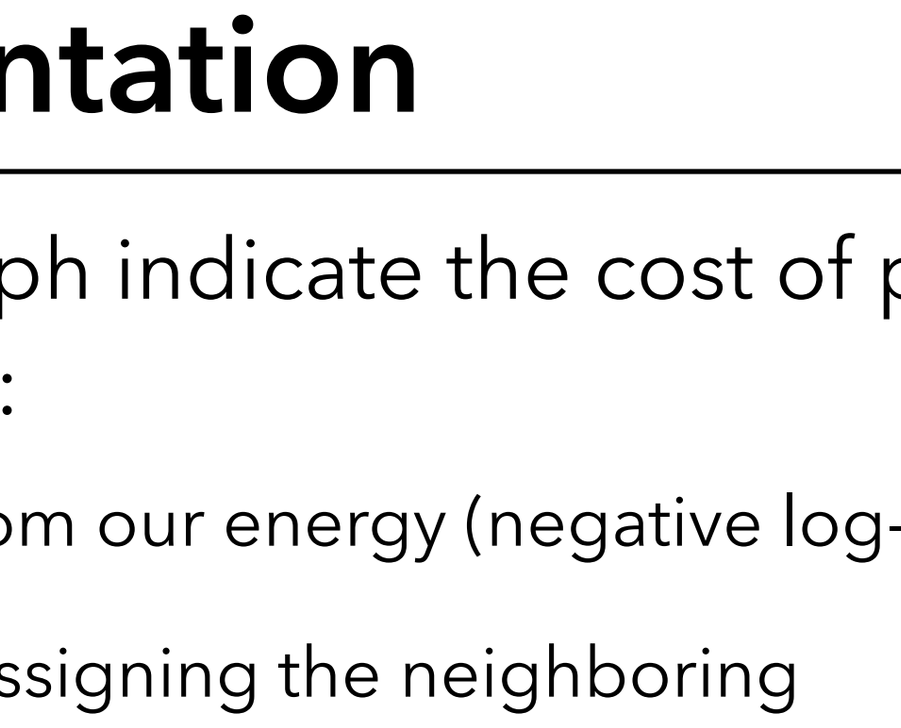
- The edges of the graph indicate the cost of pairing two nodes in the solution:
  - We get them directly from our energy (negative log-posterior).

Cost for assigning the neighboring pixels to **different** disparities

Disparity $d = 1$

Row of pixels

Disparity $d = 0$



Graph Cuts

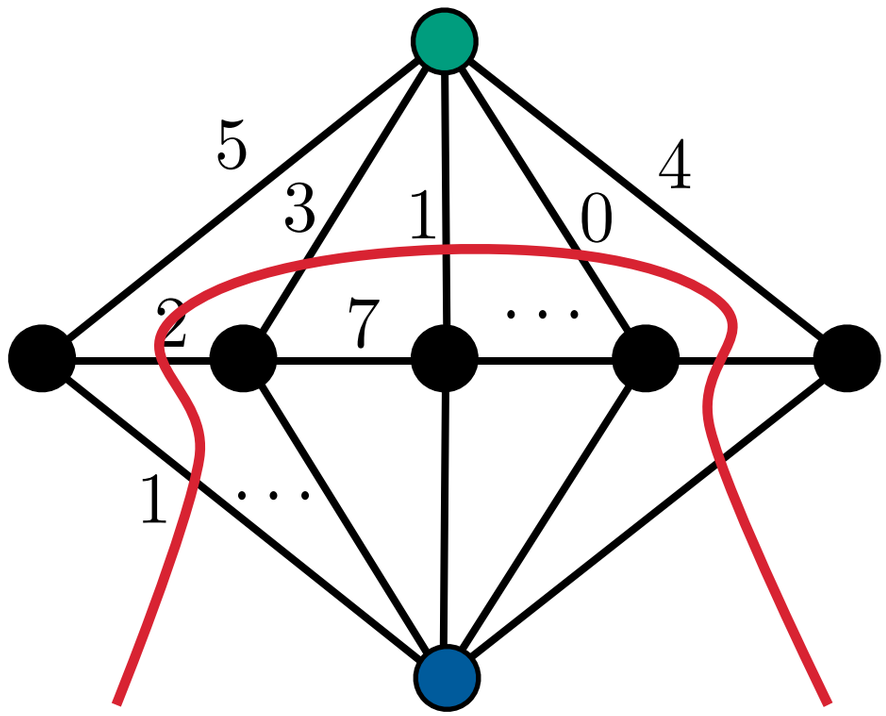
TECHNISCHE UNIVERSITÄT DARMSTADT

- MAP estimation with graph cuts [Boykov et al, 2001]:
  - We find the disparity labeling using a minimal cut of this graph.
  - The cut separates the two disparity levels (both extra nodes)

Disparity $d = 1$

Row of pixels

Disparity $d = 0$



Graph Cuts

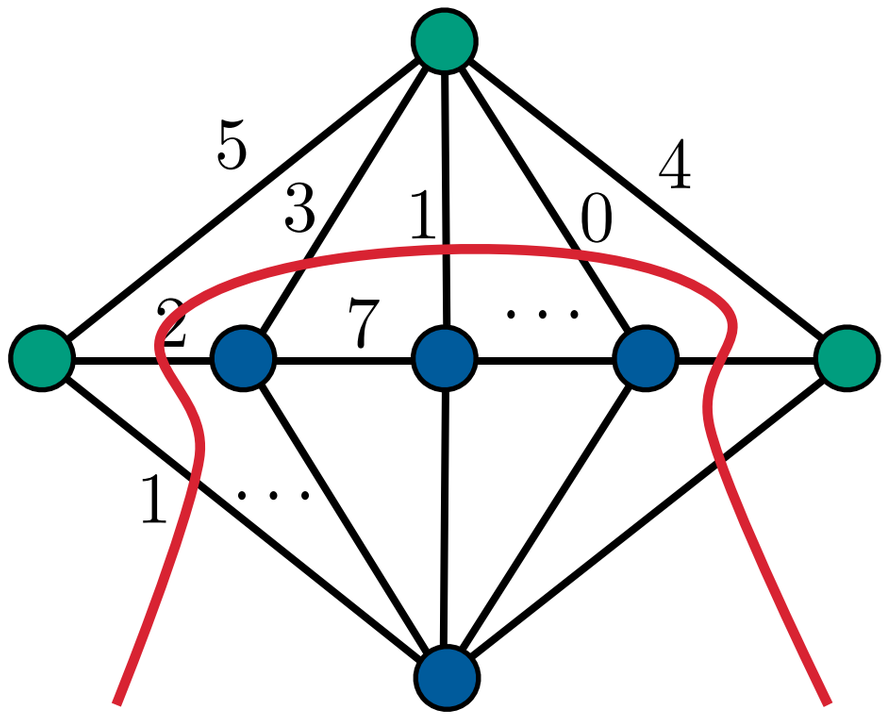
TECHNISCHE UNIVERSITÄT DARMSTADT

- MAP estimation with graph cuts [Boykov et al, 2001]:
  - We find the disparity labeling using a minimal cut of this graph.
  - The cut separates the two disparity levels (both extra nodes)
  - Assign the disparity to nodes based on which side of the cut they are on.

Disparity $d = 1$

Row of pixels

Disparity $d = 0$



Graph Cuts

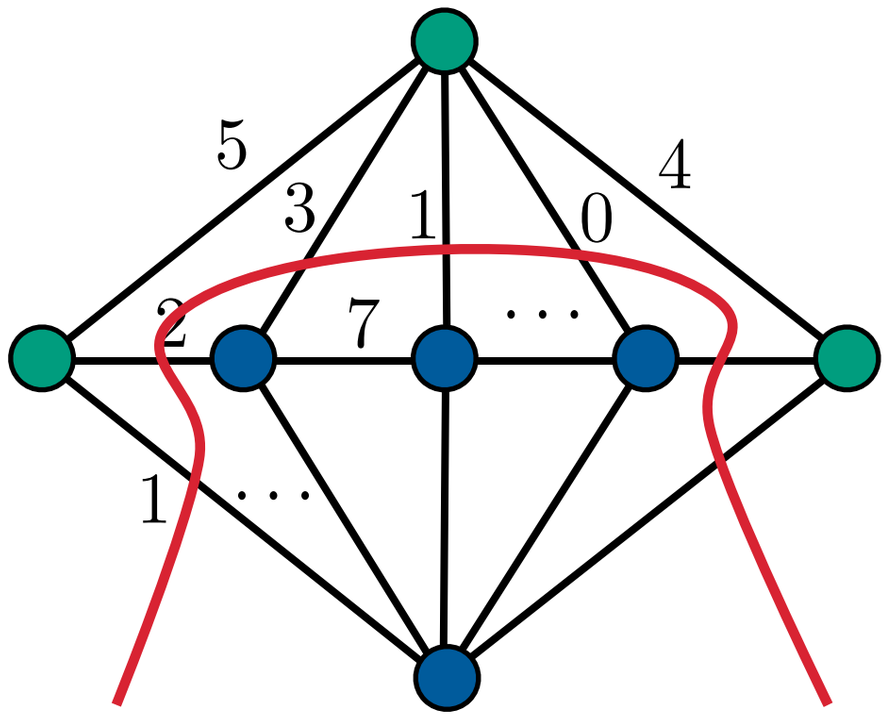
TECHNISCHE UNIVERSITÄT DARMSTADT

- MAP estimation with graph cuts [Boykov et al, 2001]:
  - How do we find a good cut?
  - We use the min-cut / max-flow algorithm (e.g. Ford-Fulkerson or more modern techniques).

Disparity $d = 1$

Row of pixels

Disparity $d = 0$



Graph Cuts Continued

TECHNISCHE UNIVERSITÄT DARMSTADT

- This technique only works directly for certain classes of energy functions, so called **submodular** ones, where the pairwise energy/cost terms satisfy:

$$g(0,0) + g(1,1) \leq g(1,0) + g(0,1)$$

  - This holds for the Potts model, and several other ones.
  - But not for all interesting cases...
- This can be extended however, so that large classes of binary energy functions from MRFs can be optimized efficiently [Kolmogorov & Rother, 07]
  - The price to pay is that only partial optimality is achieved.

Multiple Labels

TECHNISCHE UNIVERSITÄT DARMSTADT

- How can we deal with multiple labels?
- We can use the so-called **α-expansion algorithm**:
  - Yuri Boykov, Olga Veksler and Ramin Zabih: Fast Approximate Energy Minimization via Graph Cuts. IEEE Transactions on Pattern Analysis and Machine Intelligence, 23(11), 2001.
  - The algorithm uses graph cuts iteratively to find **local optima** of the energy (or log-posterior).
    - Note that the algorithm was globally optimal in the binary case.
  - In practice, the local optima found by the algorithm are very good.

α-Expansion

TECHNISCHE UNIVERSITÄT DARMSTADT

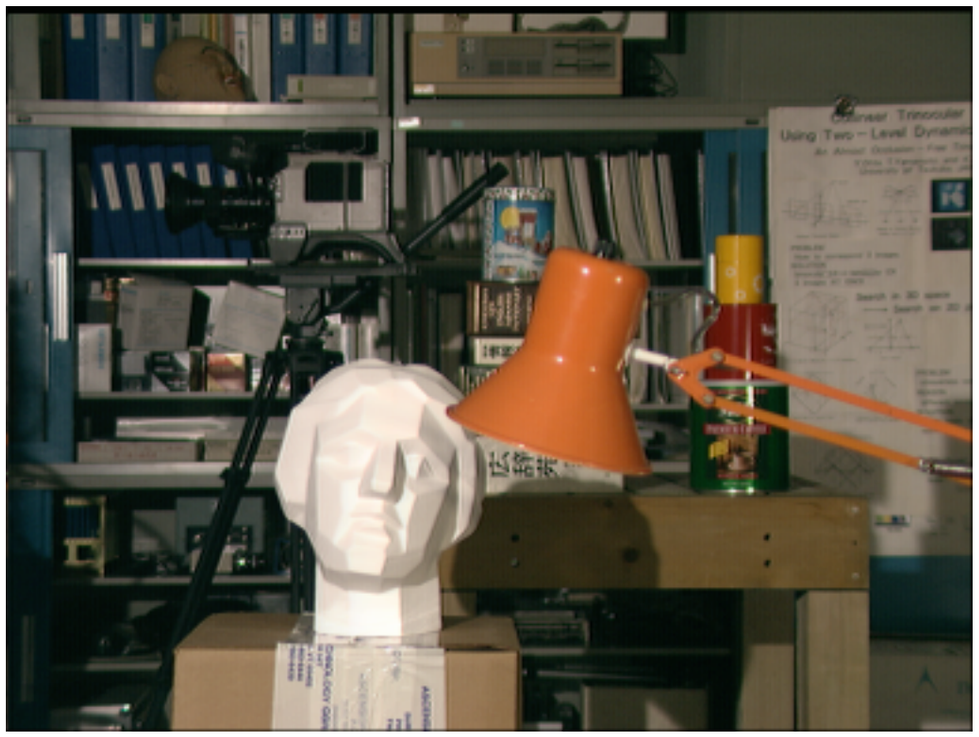
- Basic idea:
  - Initialize disparity map (e.g., all zeros)
  - Repeatedly sweep through all disparities (in a random order)
  - Treat intermediate solution as one disparity level and the current proposed disparity α as the other
  - Solve binary graph cut problem
  - Repeat until during one sweep no pixel has been updated

- Approximation guarantee:
  - No worse than factor 2 from optimal (for Potts model)

Stereo with Graph Cuts

TECHNISCHE UNIVERSITÄT DARMSTADT

- We can easily extend what we have seen to 2D graphs as we have them in stereo.
- Doing so gives very good results:

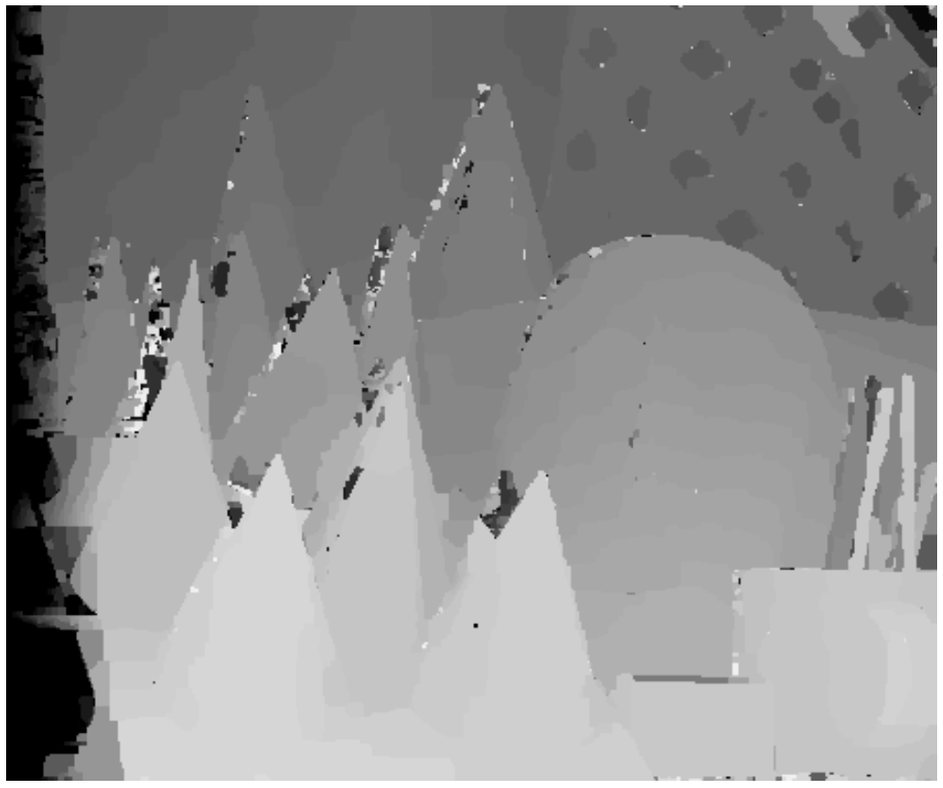


Method of [Boykov et al, 2001]

Stereo with Graph Cuts

TECHNISCHE UNIVERSITÄT DARMSTADT

- We can easily extend what we have seen to 2D graphs as we have them in stereo.
- Doing so gives very good results:



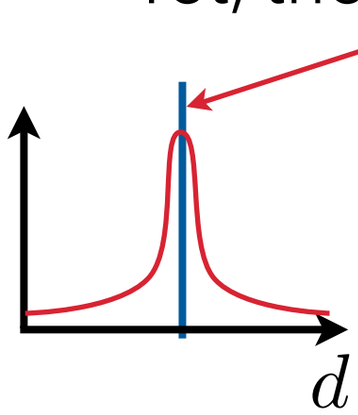
Method of [Boykov et al, 2001]

Computing Marginals

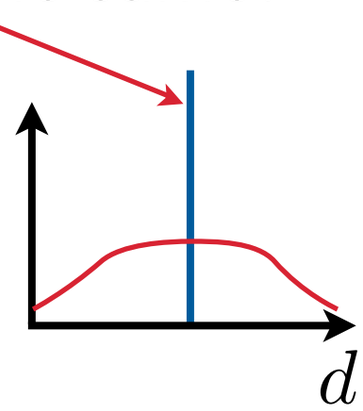
TECHNISCHE UNIVERSITÄT DARMSTADT

- Apart from computing the maximum of the posterior, it is also quite useful to be able to compute **marginal distributions** of individual (disparity) pixels:
  - We need this for learning (not covered today).
  - And it is useful for assessing **uncertainty**.
- Example: Two marginal distributions of the disparity of a single pixel

Yet, the MAP estimate is the same!



Quite certain about the result



Rather uncertain about the result

Computing Marginals

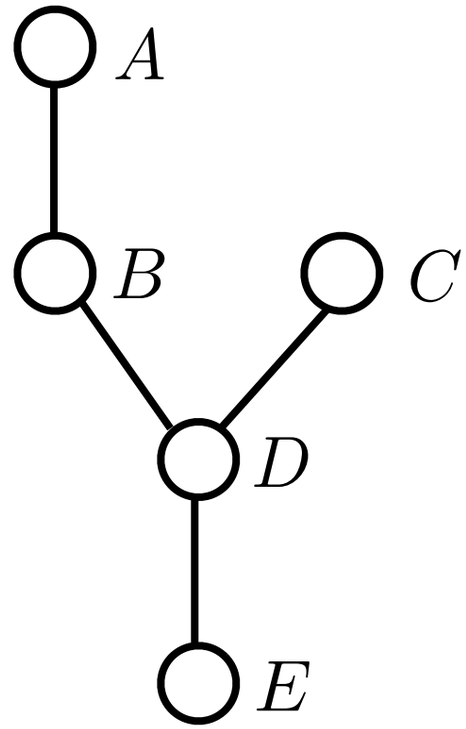
TECHNISCHE UNIVERSITÄT DARMSTADT

- Another advantage is that this allows us to compute expectations over the result:
  - We use the computed marginals to approximate the full posterior distribution and compute the expectation value pixel by pixel.
  - We can, for example, compute the minimum mean squared error (MMSE) estimate, i.e. the mean of the posterior:

$$E\left[\mathbf{d}|\mathbf{I}^0, \mathbf{I}^1\right]$$

- So how can we compute marginals?
  - The classical technique is the **sum-product algorithm** by Judea Pearl.
  - This is also called (loopy) **belief propagation** [Weiss, 1997].

Marginalization for Trees



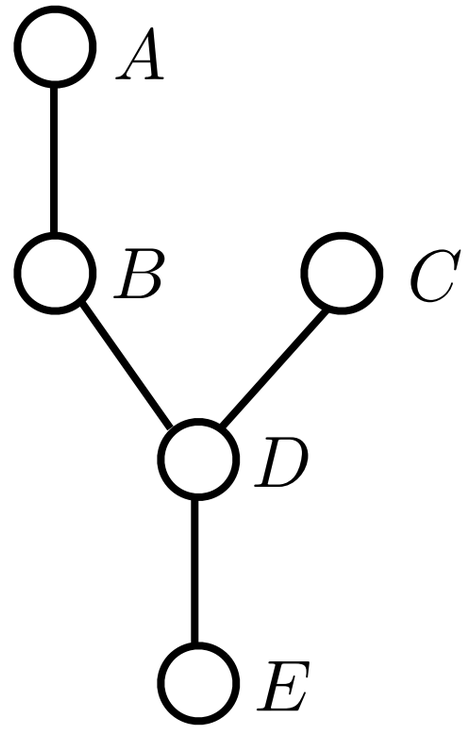
- For now, we assume a tree graph, e.g.:
- We are given the following joint distribution:

$$p(A, B, C, D, E) = \frac{1}{Z} f_1(A, B) \cdot f_2(B, D) \cdot f_3(C, D) \cdot f_4(D, E)$$

- We, for example, want to know the marginal $p(E)$

Marginalization for Trees

- Marginalize out all other variables by summing over them.
- Then rearrange terms:



$$
\begin{aligned}
p(E) &= \sum_A \sum_B \sum_C \sum_D p(A,B,C,D,E) \\
&= \sum_A \sum_B \sum_C \sum_D \frac{1}{Z} f_1(A,B) \cdot f_2(B,D) \cdot f_3(C,D) \cdot f_4(D,E) \\
&= \frac{1}{Z} \left(\sum_D f_4(D,E) \cdot \left(\sum_C f_3(C,D) \right) \cdot \left(\sum_B f_2(B,D) \cdot \left(\sum_A f_1(A,B) \right) \right) \right)
\end{aligned}
$$

Marginalization with Messages

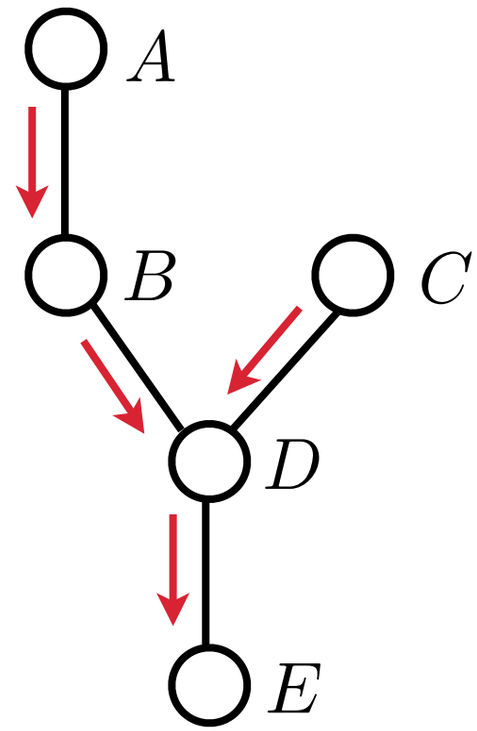
TECHNISCHE UNIVERSITÄT DARMSTADT

- We formalize the computation using the following **messages**:

$$m_{A\to B}(B) = \sum_A f_1(A,B) \qquad m_{C\to D}(D) = \sum_C f_3(C,D)$$

$$m_{B\to D}(D) = \sum_B f_2(B,D)\cdot m_{A\to B}(B)$$

$$m_{D\to E}(E) = \sum_D f_4(D,E)\cdot m_{B\to D}(D)\cdot m_{C\to D}(D)$$



$$\begin{aligned} p(E) &= \sum_A \sum_B \sum_C \sum_D p(A,B,C,D,E) \\ &= \sum_A \sum_B \sum_C \sum_D \frac{1}{Z} f_1(A,B)\cdot f_2(B,D)\cdot f_3(C,D)\cdot f_4(D,E) \\ &= \frac{1}{Z}\left(\sum_D f_4(D,E)\cdot\left(\sum_C f_3(C,D)\right)\cdot\left(\sum_B f_2(B,D)\cdot\left(\sum_A f_1(A,B)\right)\right)\right) \end{aligned}$$

Marginalization with Messages

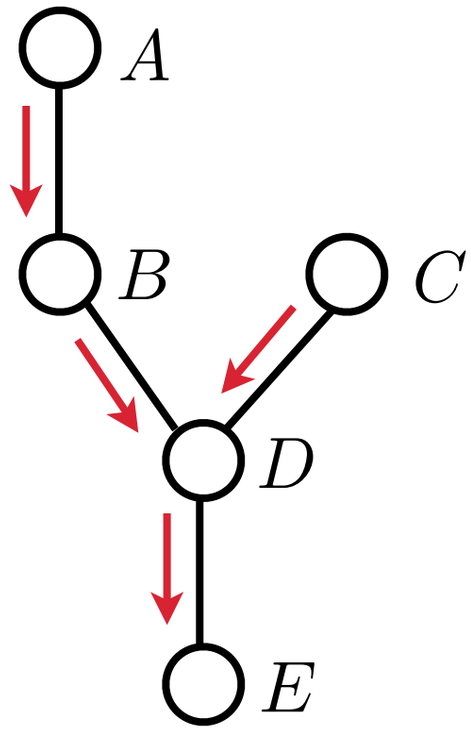
TECHNISCHE UNIVERSITÄT DARMSTADT

- We use the messages to express the marginalization:

$$m_{A\to B}(B) = \sum_A f_1(A,B) \qquad m_{C\to D}(D) = \sum_C f_3(C,D)$$

$$m_{B\to D}(D) = \sum_B f_2(B,D)\cdot m_{A\to B}(B)$$

$$m_{D\to E}(E) = \sum_D f_4(D,E)\cdot m_{B\to D}(D)\cdot m_{C\to D}(D)$$



$$\begin{aligned} p(E) &= \frac{1}{Z}\left(\sum_D f_4(D,E)\cdot\left(\sum_C f_3(C,D)\right)\cdot\left(\sum_B f_2(B,D)\cdot\left(\sum_A f_1(A,B)\right)\right)\right) \\ &= \frac{1}{Z}\left(\sum_D f_4(D,E)\cdot\left(\sum_C f_3(C,D)\right)\cdot\left(\sum_B f_2(B,D)\cdot m_{A\to B}(B)\right)\right) \end{aligned}$$

Marginalization with Messages

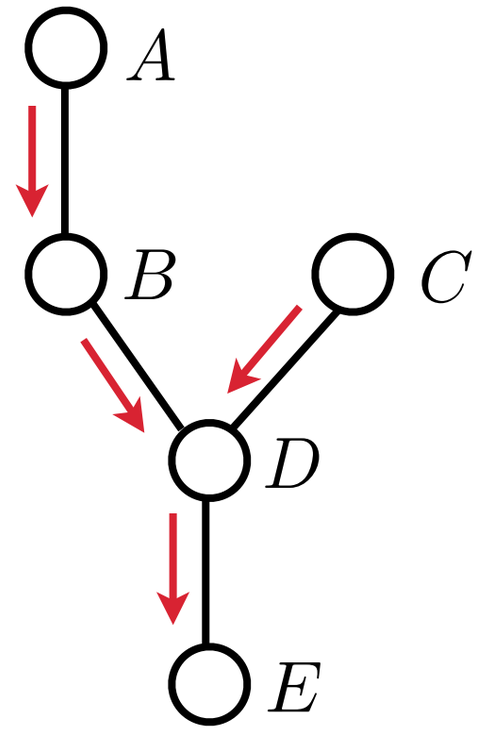
TECHNISCHE UNIVERSITÄT DARMSTADT

- We use the messages to express the marginalization:

$$m_{A\to B}(B) = \sum_A f_1(A,B) \qquad m_{C\to D}(D) = \sum_C f_3(C,D)$$

$$m_{B\to D}(D) = \sum_B f_2(B,D)\cdot m_{A\to B}(B)$$

$$m_{D\to E}(E) = \sum_D f_4(D,E)\cdot m_{B\to D}(D)\cdot m_{C\to D}(D)$$



$$\begin{aligned} p(E) &= \frac{1}{Z}\left(\sum_D f_4(D,E)\cdot\left(\sum_C f_3(C,D)\right)\cdot\left(\sum_B f_2(B,D)\cdot\left(\sum_A f_1(A,B)\right)\right)\right) \\ &= \frac{1}{Z}\left(\sum_D f_4(D,E)\cdot\left(\sum_C f_3(C,D)\right)\cdot m_{B\to D}(D)\right) \end{aligned}$$

Marginalization with Messages

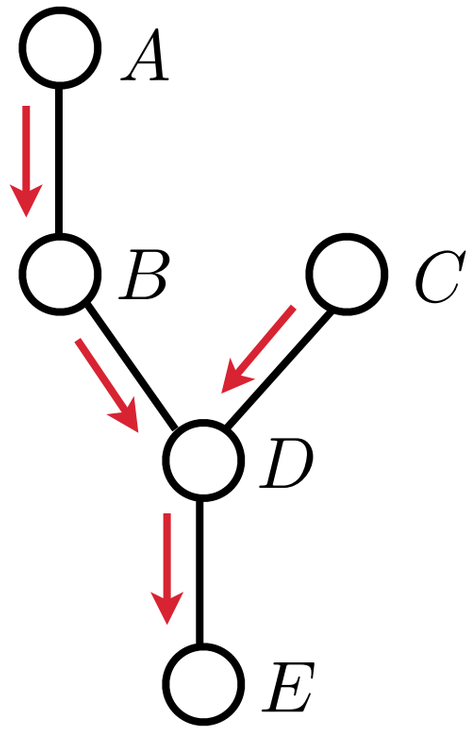
TECHNISCHE UNIVERSITÄT DARMSTADT

- We use the messages to express the marginalization:

$$m_{A\to B}(B) = \sum_A f_1(A,B) \qquad m_{C\to D}(D) = \sum_C f_3(C,D)$$

$$m_{B\to D}(D) = \sum_B f_2(B,D)\cdot m_{A\to B}(B)$$

$$m_{D\to E}(E) = \sum_D f_4(D,E)\cdot m_{B\to D}(D)\cdot m_{C\to D}(D)$$



$$\begin{aligned} p(E) &= \frac{1}{Z}\left(\sum_D f_4(D,E)\cdot\left(\sum_C f_3(C,D)\right)\cdot\left(\sum_B f_2(B,D)\cdot\left(\sum_A f_1(A,B)\right)\right)\right) \\ &= \frac{1}{Z}\left(\sum_D f_4(D,E)\cdot m_{C\to D}(D)\cdot m_{B\to D}(D)\right) \end{aligned}$$

Marginalization with Messages

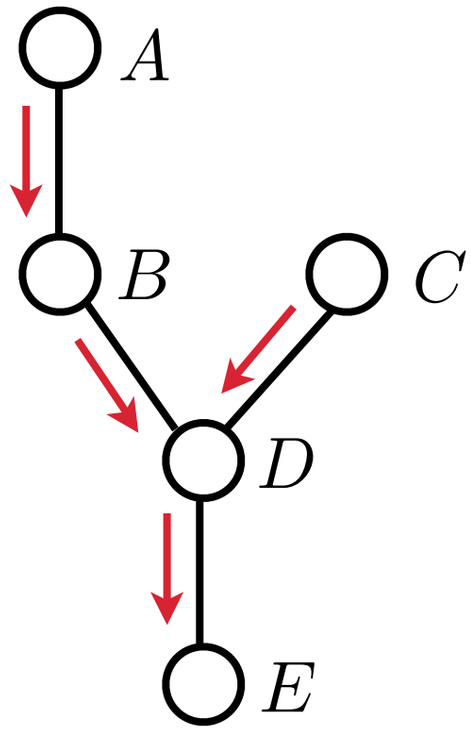
TECHNISCHE UNIVERSITÄT DARMSTADT

- We use the messages to express the marginalization:

$$m_{A\to B}(B) = \sum_A f_1(A,B) \qquad m_{C\to D}(D) = \sum_C f_3(C,D)$$

$$m_{B\to D}(D) = \sum_B f_2(B,D)\cdot m_{A\to B}(B)$$

$$m_{D\to E}(E) = \sum_D f_4(D,E)\cdot m_{B\to D}(D)\cdot m_{C\to D}(D)$$



$$\begin{aligned} p(E) &= \frac{1}{Z}\left(\sum_D f_4(D,E)\cdot\left(\sum_C f_3(C,D)\right)\cdot\left(\sum_B f_2(B,D)\cdot\left(\sum_A f_1(A,B)\right)\right)\right) \\ &= \frac{1}{Z} m_{D\to E}(E) \end{aligned}$$

Marginalization for Trees

TECHNISCHE UNIVERSITÄT DARMSTADT

- We can **generalize** this for all tree graphs.
  - First, root the tree at the variable that we want to compute the marginal of.
  - Start computing messages at the leaves.
  - Then compute the messages for all nodes for which all incoming messages have already been computed.
  - Repeat until we reach the root.
- If we want to compute the marginals for all possible nodes (roots), we can actually reuse some of the messages.
  - The computational expense in linear in the number of nodes.

The Sum-Product Algorithm

TECHNISCHE UNIVERSITÄT DARMSTADT

- Formalize this as follows [Pearl].
- Nodes of the graph: $x_i, \quad i = 1, \ldots, N$
- Neighboring nodes of x_i in the graph: $x_j, \quad j \in \mathcal{N}(i)$
- **Message rule:**

$$m_{i \to j}(x_j) := \sum_{x_i} f_{ij}(x_i, x_j) \prod_{k \in \mathcal{N}(i) \setminus j} m_{k \to i}(x_i)$$

- **Marginals:**

$$p(x_i) \propto \prod_{k \in \mathcal{N}(i)} m_{k \to i}(x_i)$$

Loopy Belief Propagation

TECHNISCHE UNIVERSITÄT DARMSTADT

- In many applications, for example in computer vision, the graphical models we use are “loopy”, i.e. they are not trees.
  - What can we do?
- **(Loopy) Belief Propagation (BP or LBP):**
  - Apply the sum-product algorithm even if the graph is not a tree.
  - Iteratively update messages until they converge (sometimes they don’t converge).
  - At convergence, the so-called beliefs $b(x_i)$ are an approximation of the marginals of the loopy graph:

$$p(x_i) \approx b(x_i) \propto \prod_{k \in \mathcal{N}(i)} m_{k \to i}(x_i)$$

Loopy Belief Propagation

TECHNISCHE UNIVERSITÄT DARMSTADT

- Even though this seems like a hack, there is a thorough mathematical justification for why this is a good approximation [Yedidia et al., 2001].
- Max-product belief propagation:
  - There is a slight variation of BP that we can use to approximately compute MAP estimates:

$$m_{i \to j}(x_j) := \max_{x_i} f_{ij}(x_i, x_j) \prod_{k \in \mathcal{N}(i) \setminus j} m_{k \to i}(x_i)$$

$$x_i^{\text{MAP}} \approx \arg\max_{x_i} b(x_i) = \arg\max_{x_i} \prod_{k \in \mathcal{N}(i)} m_{k \to i}(x_i)$$

Stereo Wrap-Up

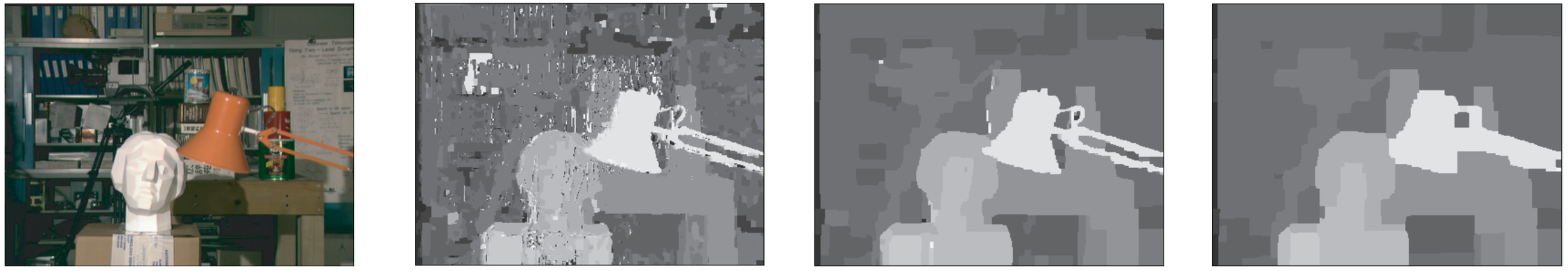
TECHNISCHE UNIVERSITÄT DARMSTADT

- What have we learned about stereo?
  - We can relate scene depth to disparity on the image plane.
  - We have to search for correspondences along scanlines (or epipolar lines in the general case) in order to determine disparity.
  - We need to impose some form of prior knowledge of spatial regularity to obtain good results.
  - We can impose prior knowledge using pairwise Markov random fields.
  - Probabilistic inference is difficult in loopy MRFs.
  - We can nevertheless do approximate inference using graph cuts or belief propagation.

Limitations of the Approach

TECHNISCHE UNIVERSITÄT DARMSTADT

- The simple MRF model does not properly deal with occluded and disoccluded areas.
  - Can introduce occlusion reasoning.
- We need to properly tune the parameters:



Different parameters of the prior lead to different solutions

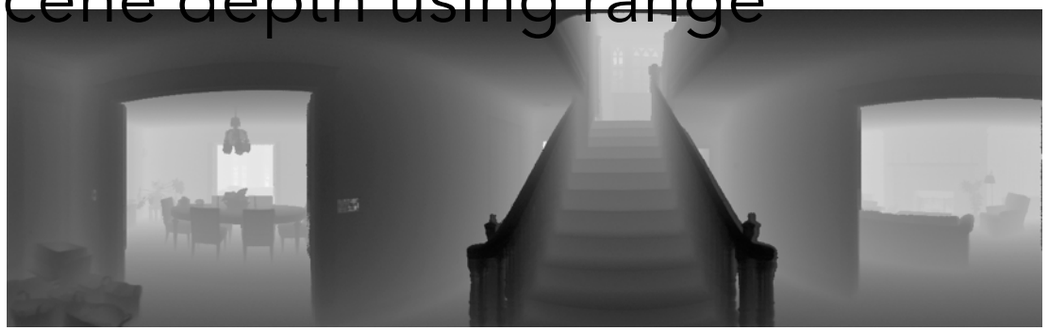
- Hand tuning is tedious or even too hard to do when there are many parameters.
  - We can instead learn the parameters.

Images from [Zhang & Seitz]

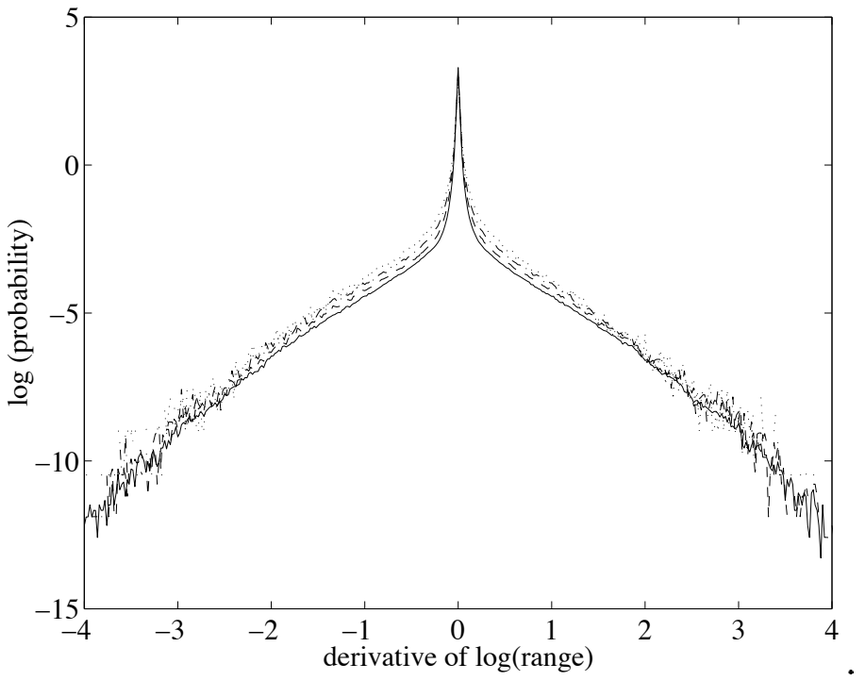
Limitations of the Approach

TECHNISCHE UNIVERSITÄT DARMSTADT

- The Potts prior is only a poor model of the statistics of disparity in real scenes.
  - People have studied the statistics of scene depth using range images.



  - The derivative statistics of range images have a very characteristic



Because the difference of neighboring pixels that we use for modeling is an approximation of the image derivatives, this has implications for modeling disparity priors.

Did we meet our goals?

TECHNISCHE UNIVERSITÄT DARMSTADT

- We wanted consistency within and between scanlines.
  - *Yes, the MRF prior provides that.*
- We wanted a model of consistency that is well supported by the properties of the real world, i.e. by real scene depth.
  - *The Markov assumption is a restriction, but still quite reasonable.*
  - *The Potts model is restrictive, but also very simple to deal with. Better models for the factors are not that hard to devise.*
- We wanted a model that is computationally manageable.
  - *We can do (high-quality) approximate inference in polynomial time.*
- We wanted to find a model of consistency that not only works for stereo, but also for other applications.

Where can we go from here?

TECHNISCHE UNIVERSITÄT DARMSTADT

- We have also seen that our model is only a simplification of the problem:
  - We could define a more realistic likelihood model.
  - In particular, we could reason about occlusions and disocclusions.
  - We could make the prior more realistic:
    - For example by improving the potential function.
    - Or we could make the graph more expressive, for example using bigger neighborhoods, e.g. to model the slant of the surfaces
- Etc.

How do we know what works?

TECHNISCHE UNIVERSITÄT DARMSTADT

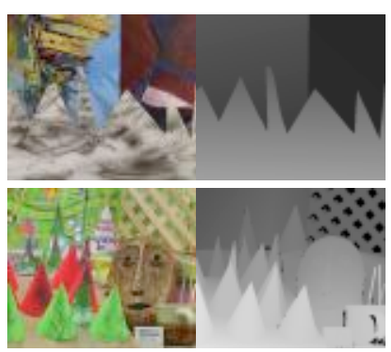
- These are all good ideas, but how do we know if we actually improved performance?
- We need stereo data sets with ground truth disparity:
  - Then we can compare the method's output with the ground truth.
  - Middlebury Stereo Benchmark [Scharstein & Szeliski, 2002]:

vision.middlebury.edu

stereo · mview · MRF · flow

Stereo | Evaluation · Datasets · Code · Submit

Middlebury Stereo Datasets



2001 datasets - 6 datasets of piecewise planar scenes [1]
(Sawtooth, Venus, Bull, Poster, Barn1, Barn2)

2003 datasets - 2 datasets with ground truth obtained using structured light [2]
(Cones, Teddy)

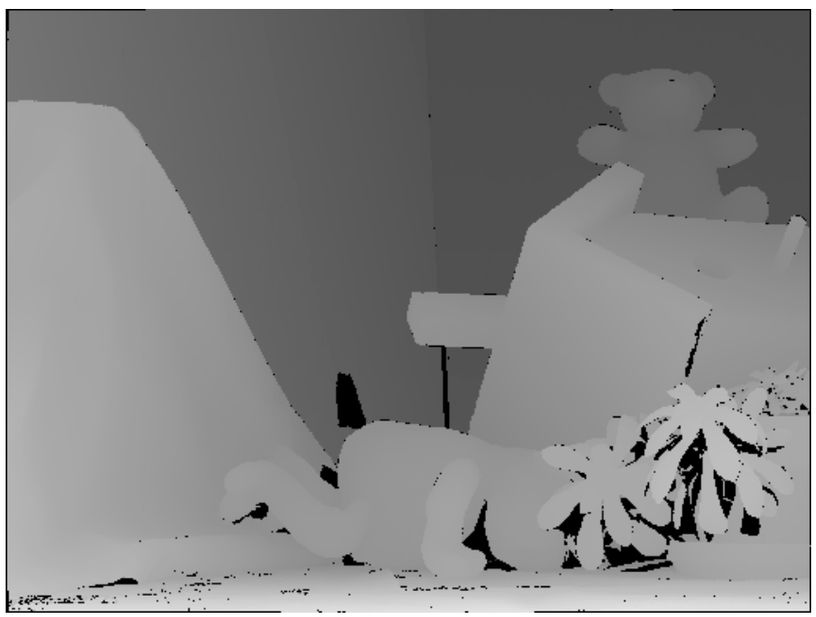
Stereo Ground Truth

TECHNISCHE UNIVERSITÄT DARMSTADT

- Image pairs with disparity pairs (one shown):

left

right

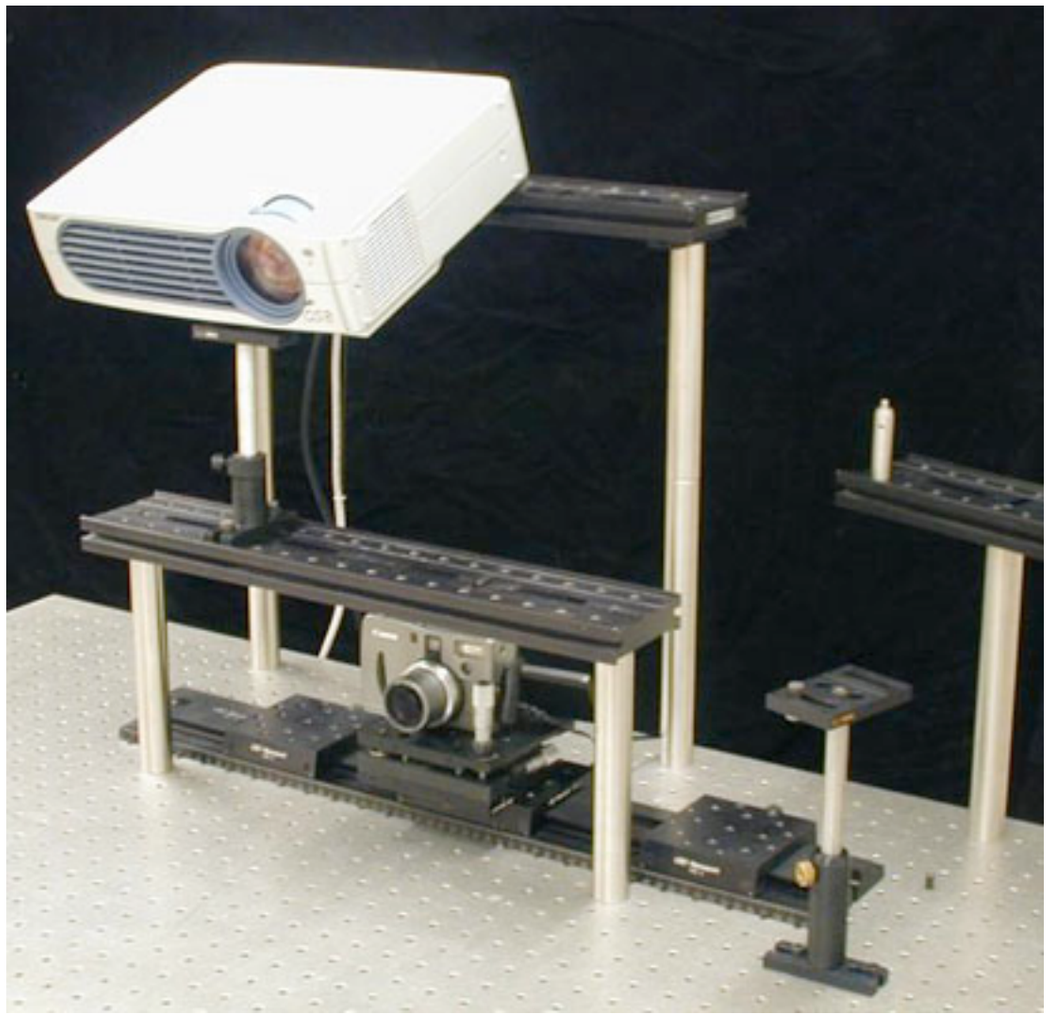


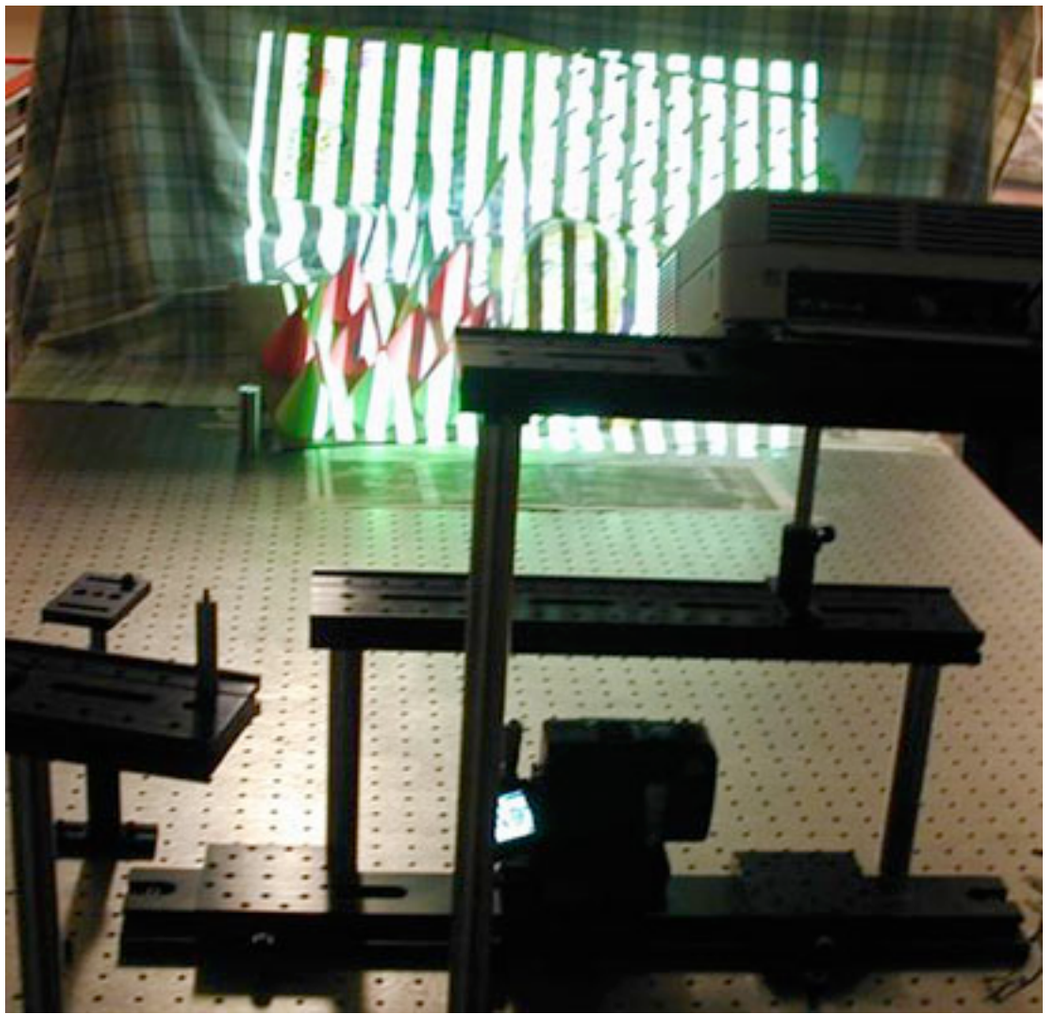
[Scharstein & Szeliski, 2002]

How was the ground truth obtained?

TECHNISCHE UNIVERSITÄT DARMSTADT

- **Structured light:**
  - Project various stripe patterns on the scene and observe how they deform.

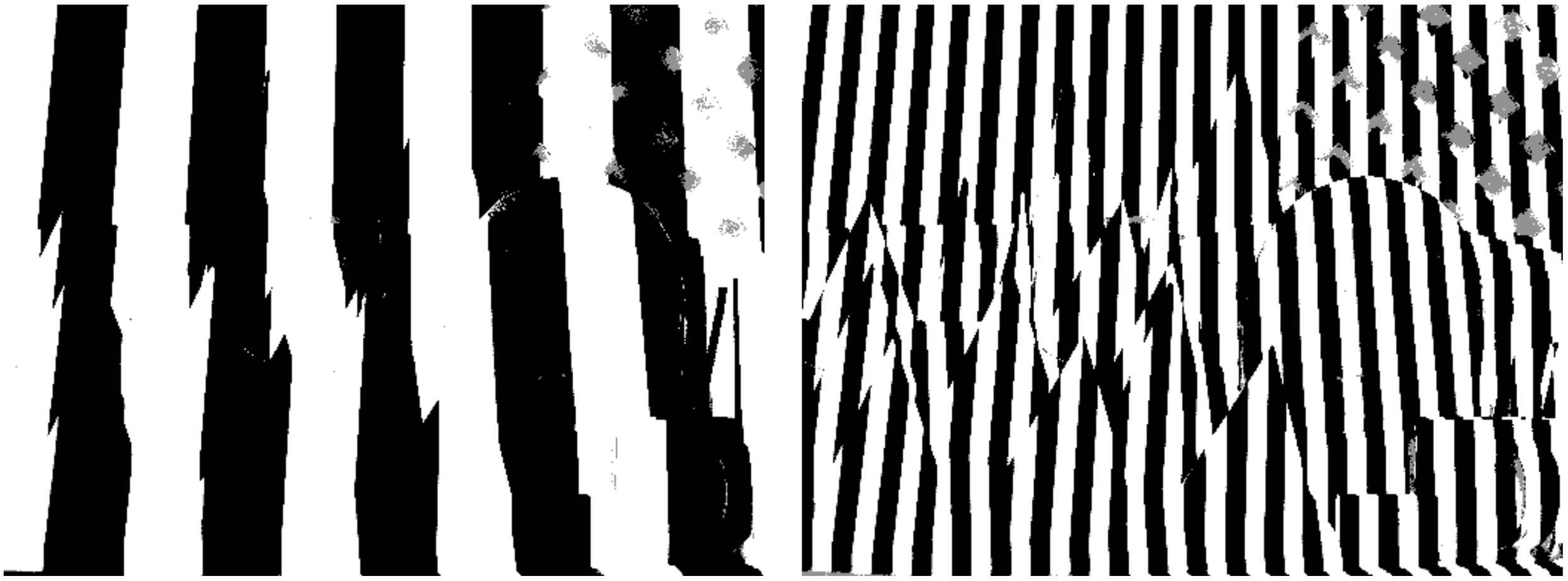




How was the ground truth obtained?

TECHNISCHE UNIVERSITÄT DARMSTADT

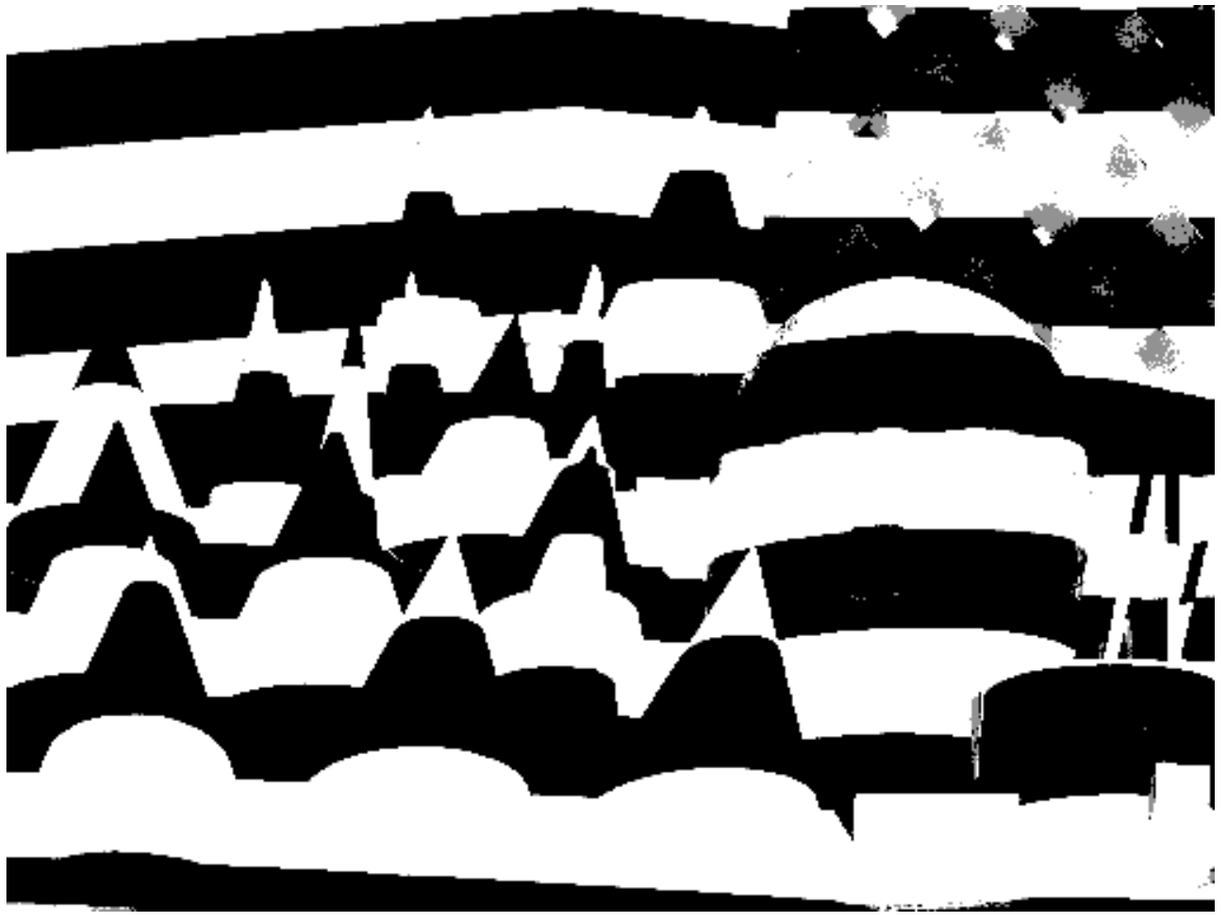
- Structured light:
  - Project various stripe patterns on the scene and observe how they deform.

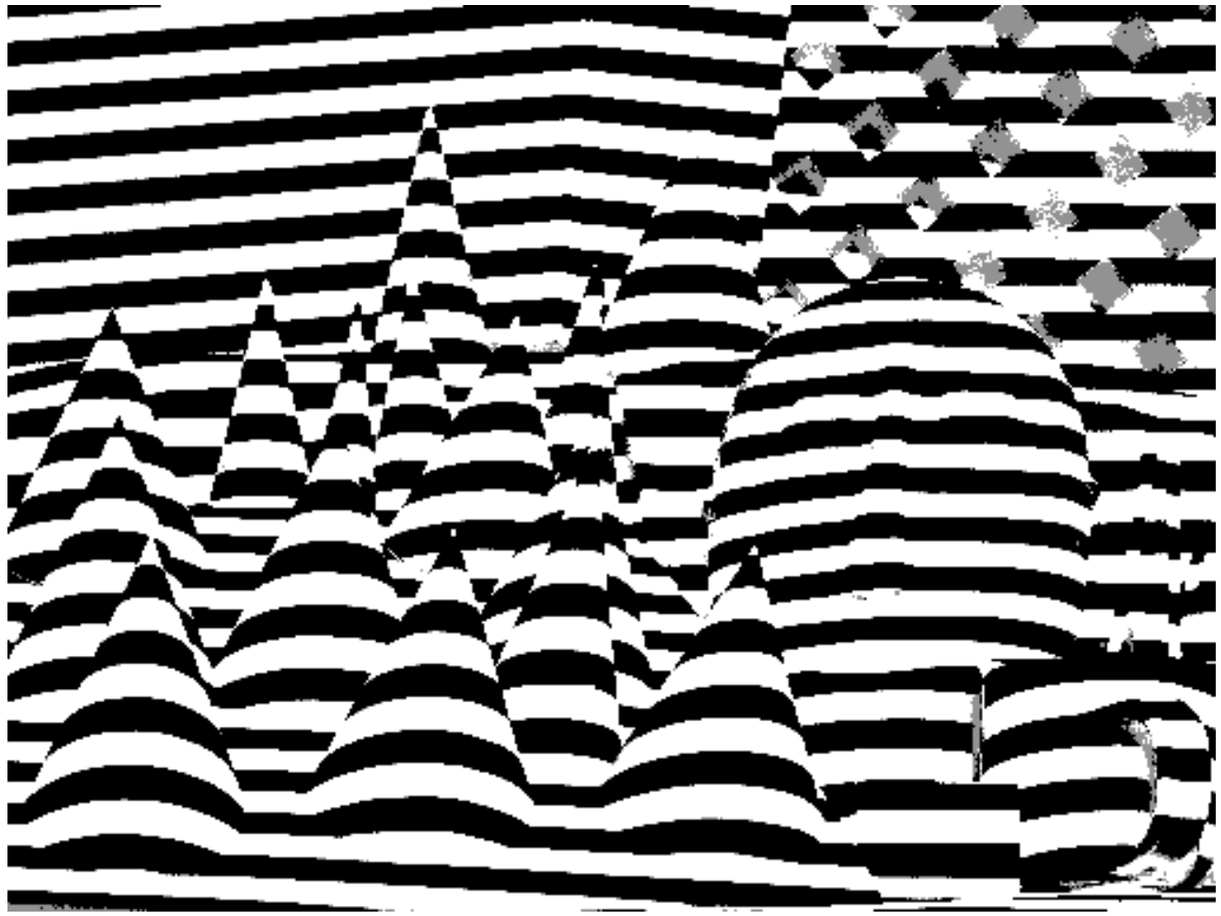


How was the ground truth obtained?

TECHNISCHE UNIVERSITÄT DARMSTADT

- Structured light:
  - Project various stripe patterns on the scene and observe how they deform.

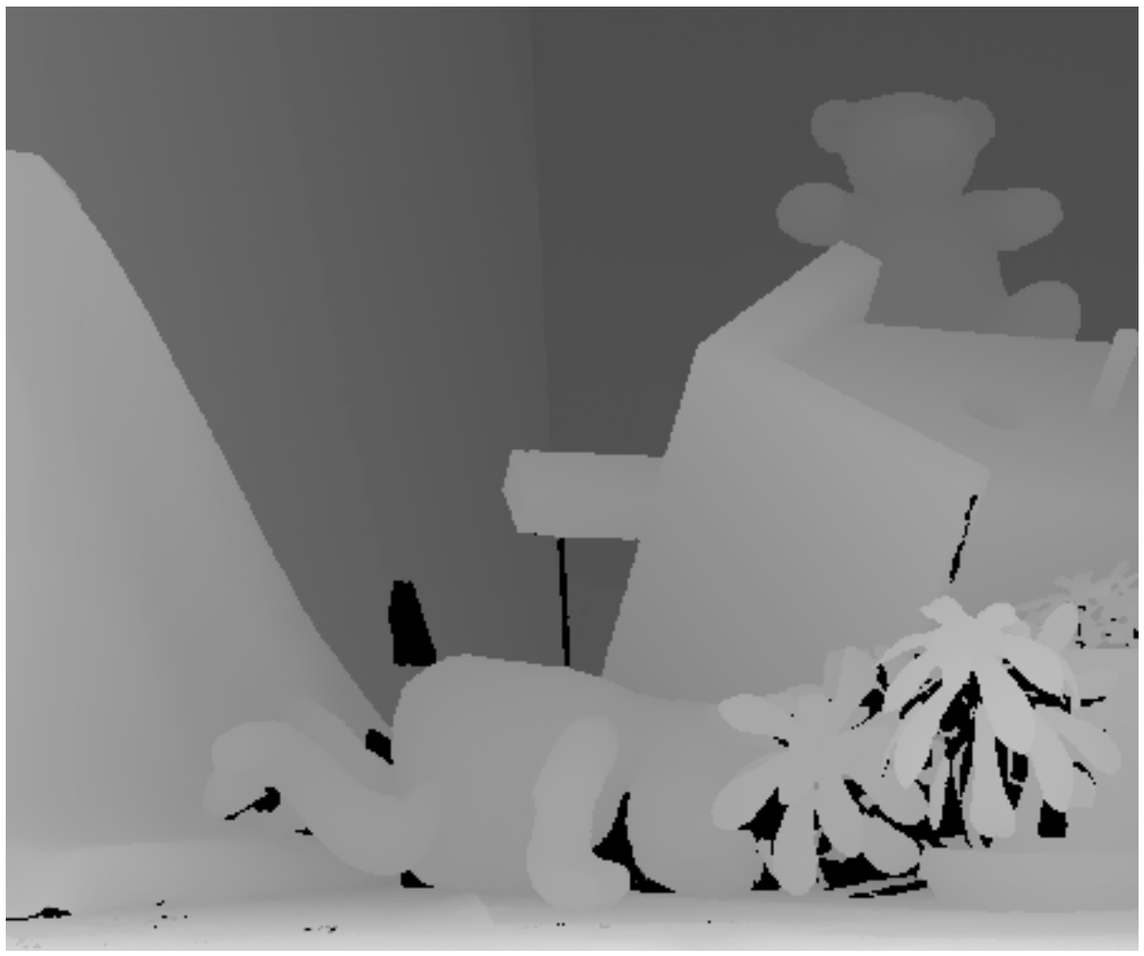




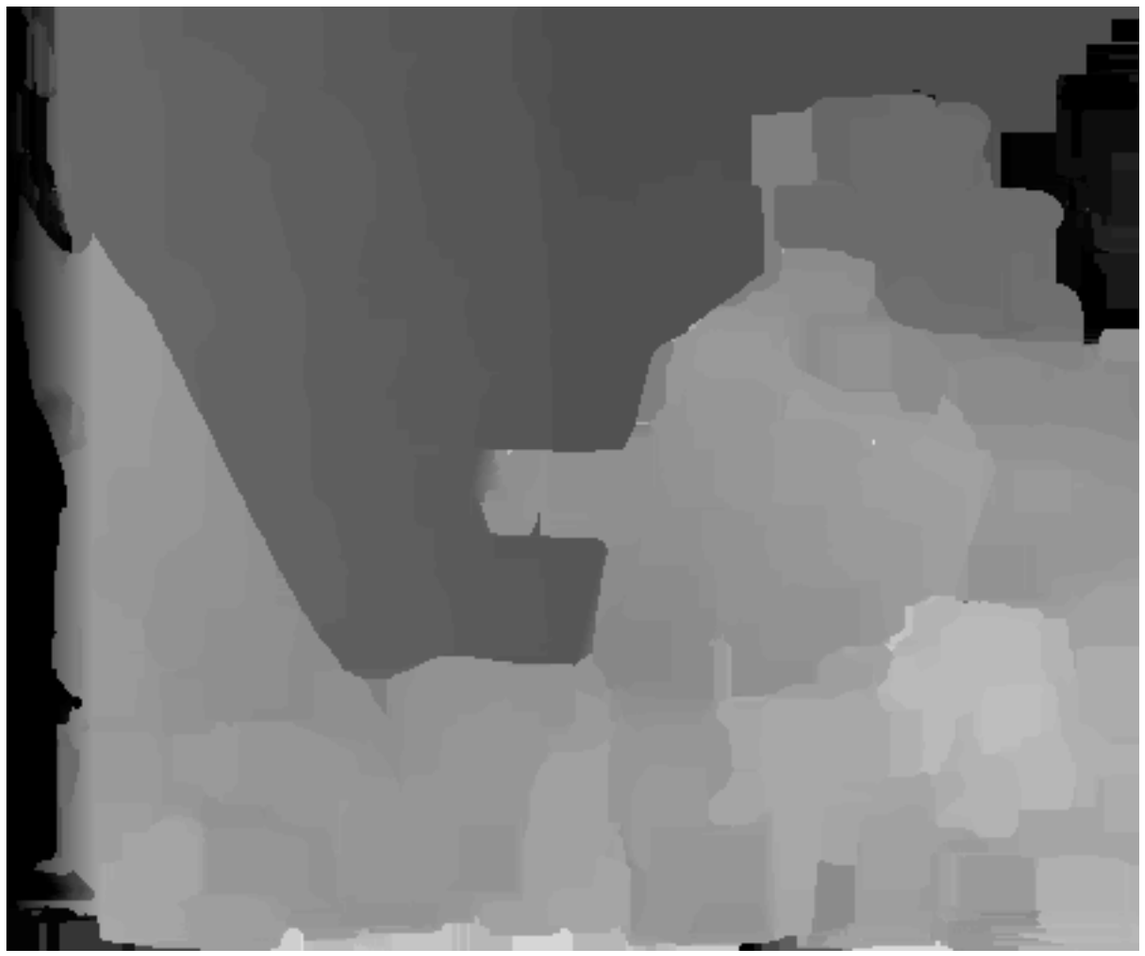
How do various techniques perform?

TECHNISCHE UNIVERSITÄT DARMSTADT

- Window-based matching with SSD (21 x 21):



Ground truth



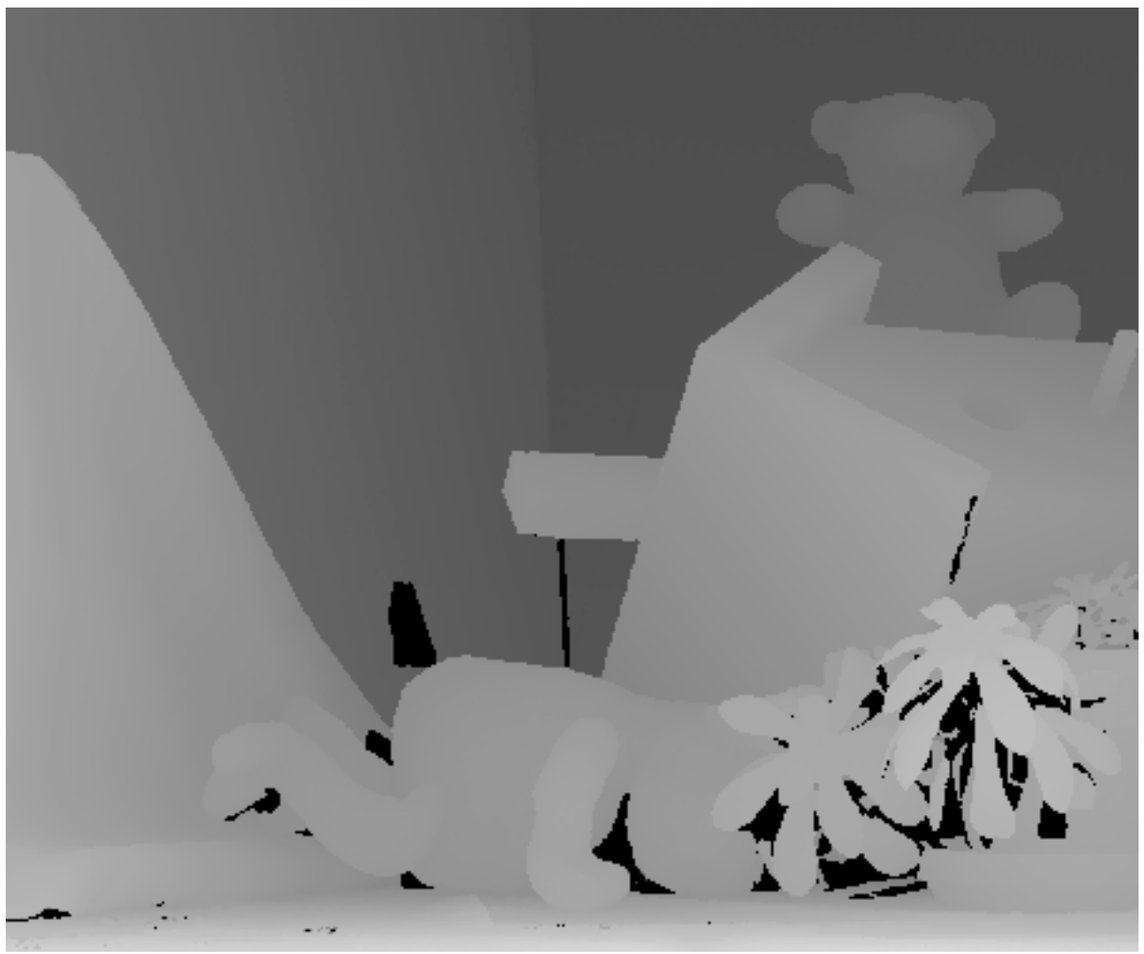
Estimated disparity

10.8% of non-occluded pixels have an error of 1 pixel or more

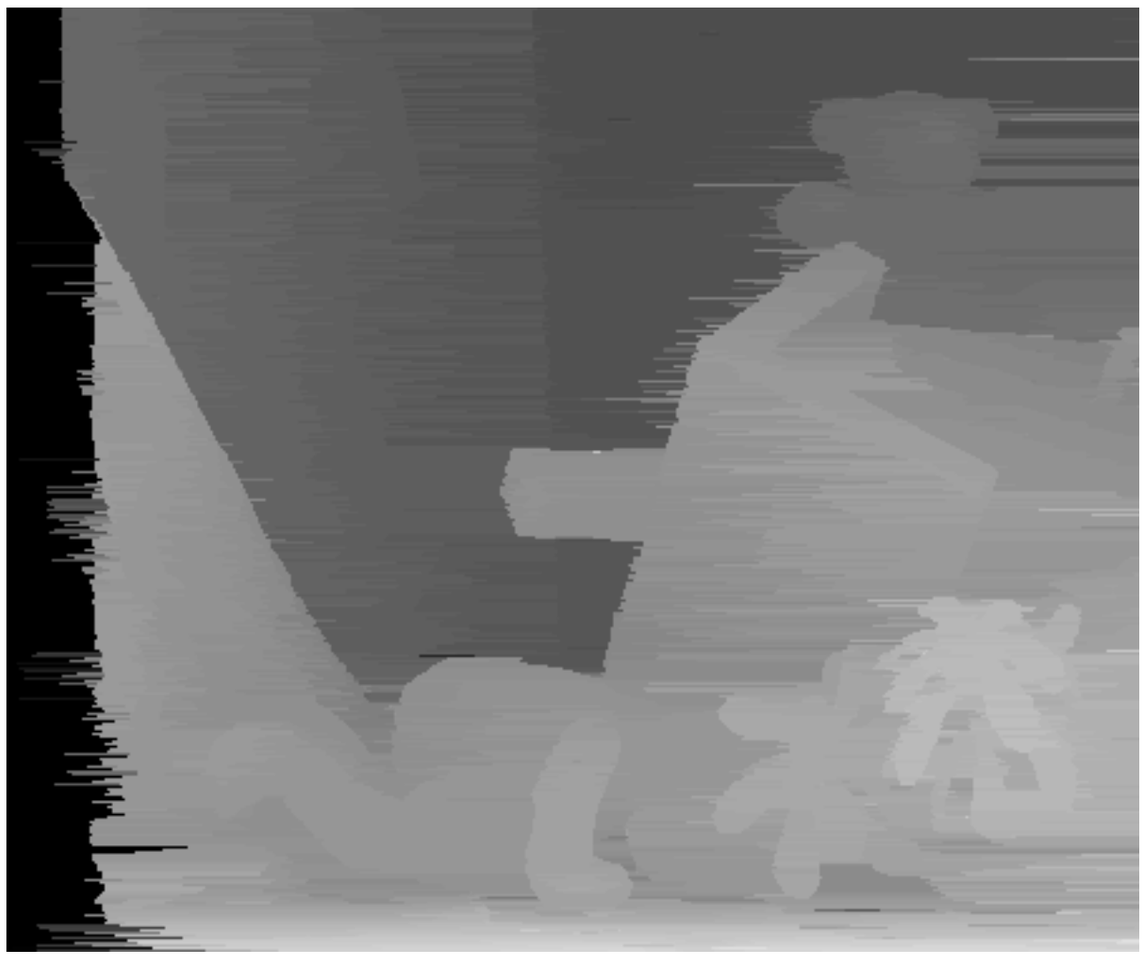
How do various techniques perform?

TECHNISCHE UNIVERSITÄT DARMSTADT

- Dynamic Programming (refined):



Ground truth



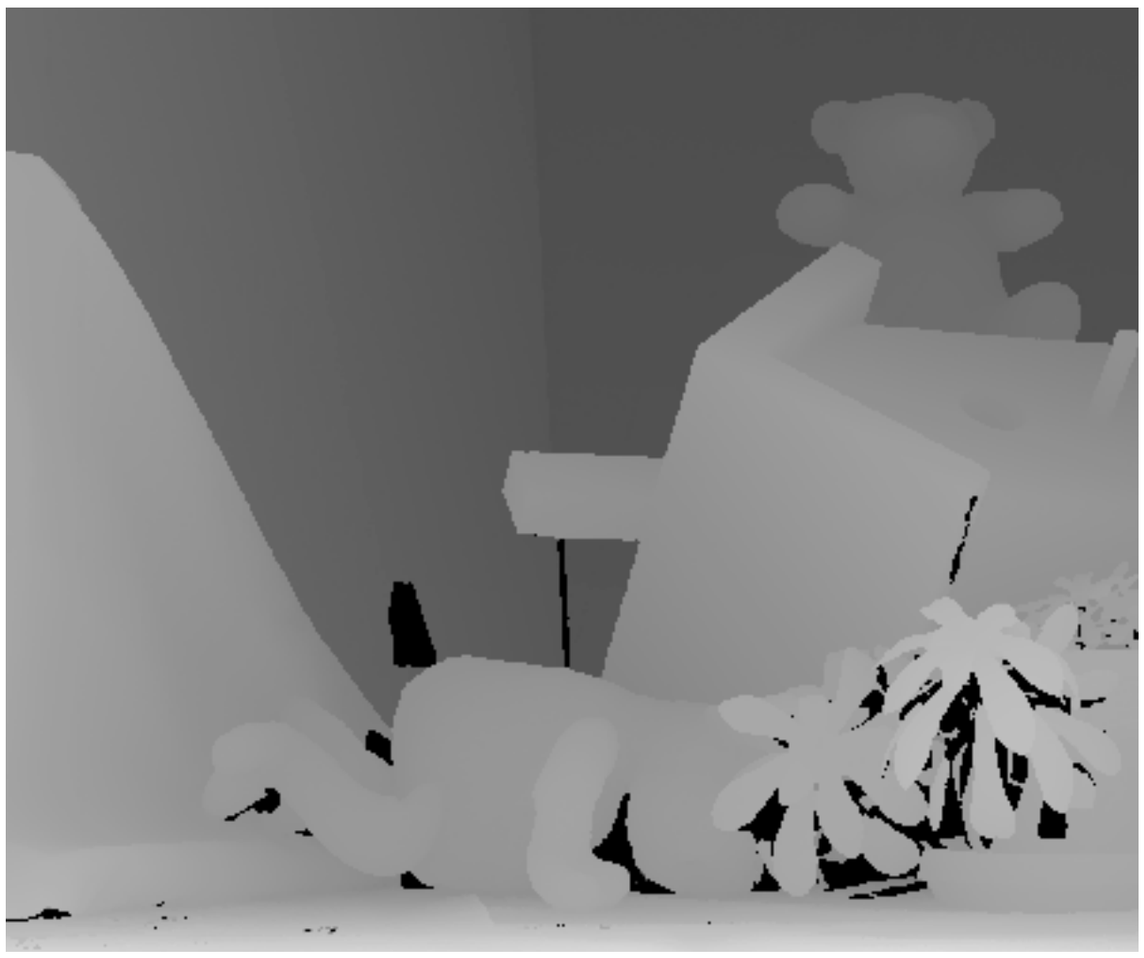
Estimated disparity

10.5% of non-occluded pixels have an error of 1 pixel or more

How do various techniques perform?

TECHNISCHE UNIVERSITÄT DARMSTADT

- MRF + Graph Cuts:



Ground truth

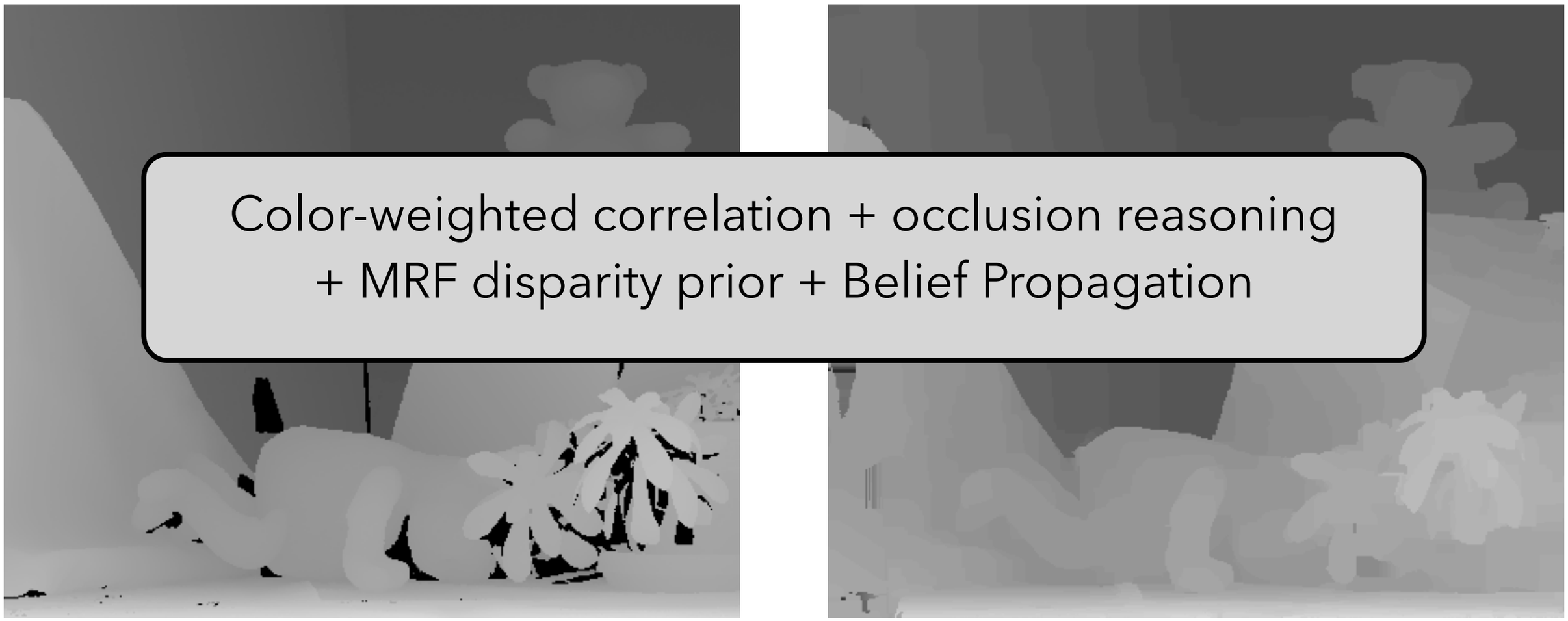
Estimated disparity

7.7% of non-occluded pixels have an error of 1 pixel or more

How do various techniques perform?

TECHNISCHE UNIVERSITÄT DARMSTADT

- **Highly performing method:** Double BP [Yang et al. 06]



Ground truth

Estimated disparity

3.55% of non-occluded pixels have an error of 1 pixel or more

Next Topic

TECHNISCHE UNIVERSITÄT DARMSTADT

- **Promise:** We will have more pictures than so far :-)
- Applications of these techniques to image processing and more.

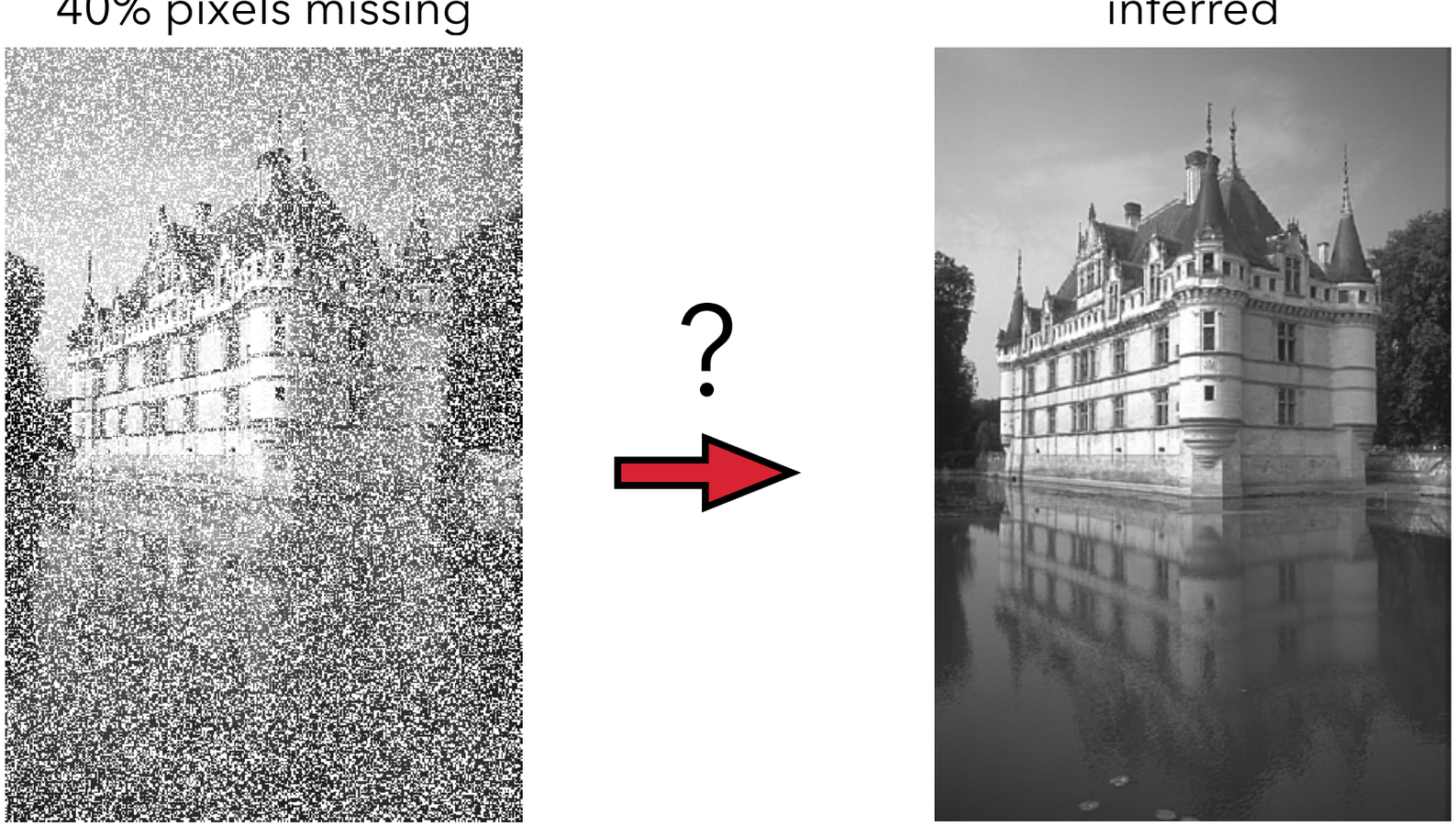


Image Restoration

07.05.2014

TECHNISCHE UNIVERSITÄT DARMSTADT

visual inference

Image Restoration

TECHNISCHE UNIVERSITÄT DARMSTADT

- Now we will shift gears and look at another problem domain: **Image restoration**
  - We will see that we can use a number of techniques that are very similar to what we have used in stereo.
- We will look at 3 applications:
  - Image denoising.
  - Image inpainting.
  - Super-resolution.

Image Denoising

TECHNISCHE UNIVERSITÄT DARMSTADT

- Essentially all digital images exhibit **image noise** to some degree.
  - Even very high quality images contain some noise.
  - Really noisy images may look like they are dominated by noise.
- Image noise is both:
  - Visually disturbing for the human viewer.
  - Distracting for vision algorithms.
- Image denoising aims at removing or reducing the amount of noise in the image.

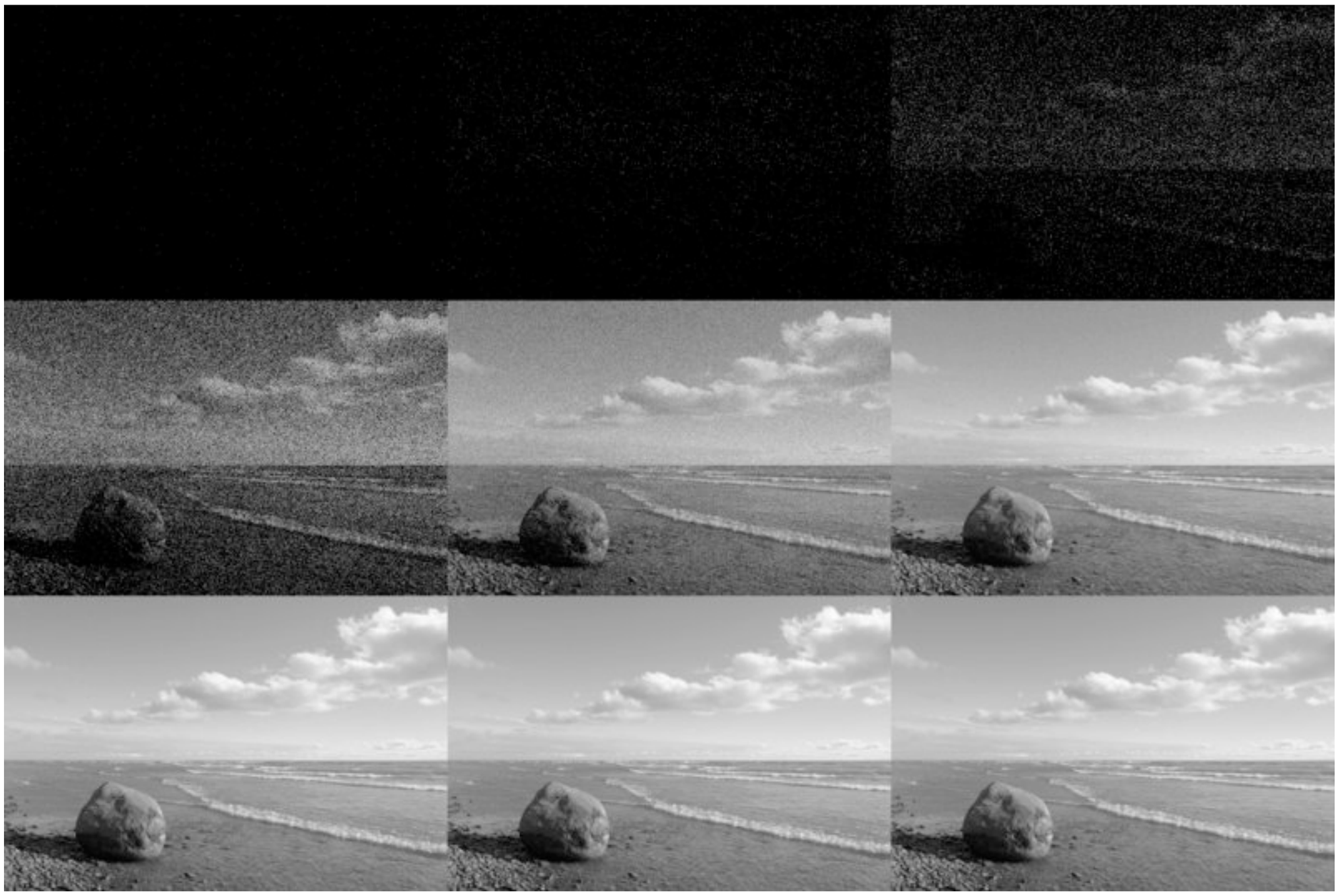
Image Noise

TECHNISCHE UNIVERSITÄT DARMSTADT

- Image noise is a very common phenomenon that can come from a variety of sources:
  - Shot noise or photon noise due to the stochastic nature of the photons arriving at the sensor.
    - Cameras actually count photons!
  - Thermal noise (“fake” photon detections).
  - Processing noise within CMOS or CCD, or in camera electronics.
  - If we use “analog” film, there are physical particles of a finite size that cause noise in a digital scan.

Shot Noise

TECHNISCHE UNIVERSITÄT DARMSTADT



Simulated shot noise (Poisson process)

From Wikipedia

Thermal Noise or “Dark Noise”

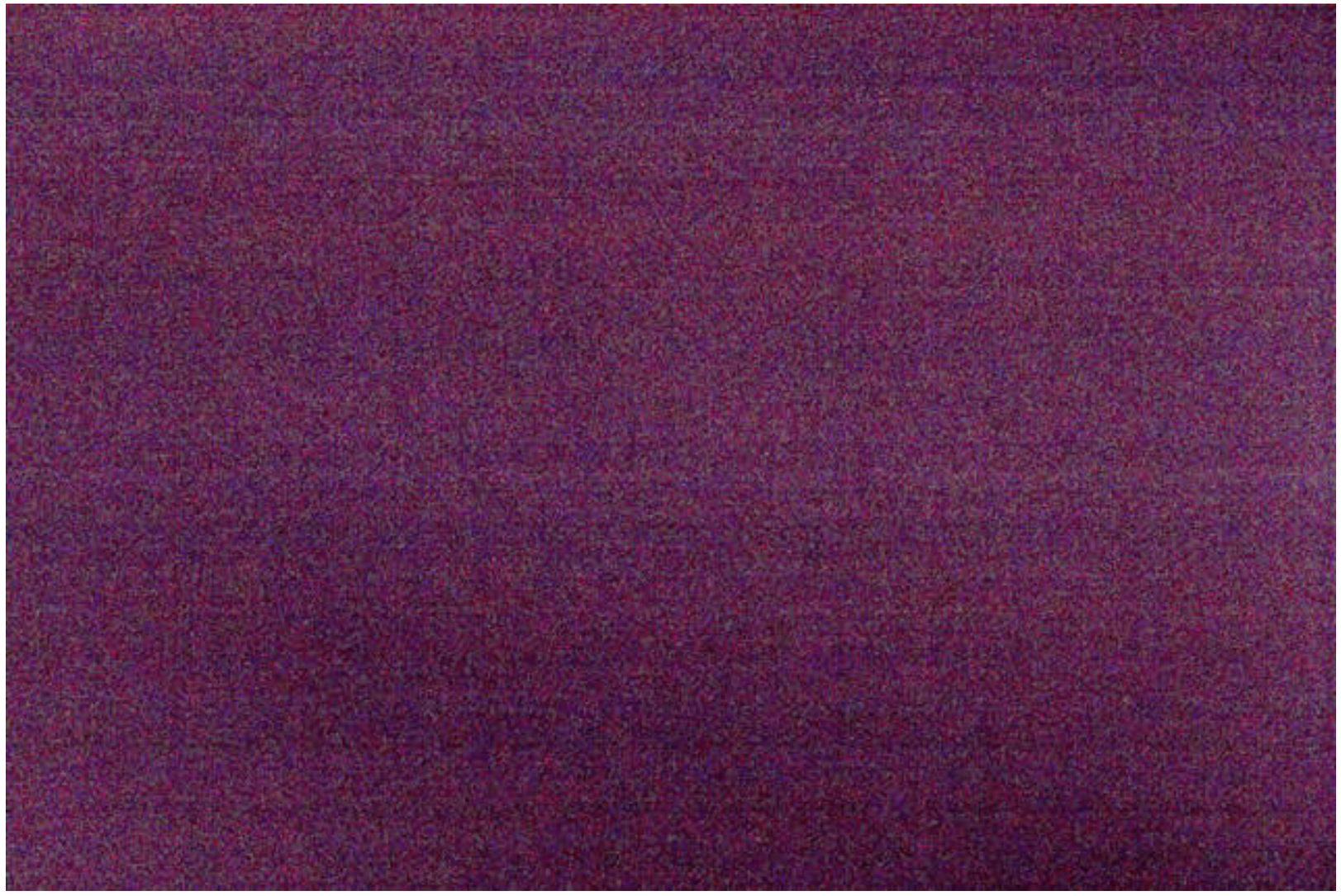
TECHNISCHE UNIVERSITÄT DARMSTADT

62 minute exposure with no incident light

From Jeff Medkeff (photo.net)

Bias Noise from Amplifiers

TECHNISCHE UNIVERSITÄT DARMSTADT



High ISO exposure (3200)

From Jeff Medkeff (photo.net)

Noise in Real Digital Photographs

TECHNISCHE UNIVERSITÄT DARMSTADT

- Combination of many different noise sources:

From dcresource.com

Film Grain

TECHNISCHE UNIVERSITÄT DARMSTADT

- If we make a digital scan of a film, we get noise from the small silver particles on the film strip:

Das Testament des Dr. Mabuse, 1933

How do we remove image noise?

TECHNISCHE UNIVERSITÄT DARMSTADT

- Classical techniques:
  - Linear filtering, e.g. Gauss filtering.
  - Median filtering
  - Wiener filtering
  - Etc.
- Modern techniques:
  - PDE-based techniques
  - Wavelet techniques
  - **MRF-based techniques**
  - Etc.