Perception, Part 3
Vision

Computer Science 6912

Department of Computer Science
Memorial University of Newfoundland

June 13, 2016
1. Visual Sensors

2. Visual Ranging

3. Rectification
Vision is often considered the most important sense, it is therefore not surprising that visual sensors have advanced so far in recent years.
Vision is often considered the most important sense, it is therefore not surprising that visual sensors have advanced so far in recent years.

Two of the most popular varieties are CCD and CMOS sensors.
Visual Sensors

- Vision is often considered the most important sense, it is therefore not surprising that visual sensors have advanced so far in recent years.
- Two of the most popular varieties are CCD and CMOS sensors.
- Both are arrays of *pixels* → light-sensitive electronic elements which convert incident light energy into charge.
Visual Sensors

- Vision is often considered the most important sense, it is therefore not surprising that visual sensors have advanced so far in recent years
- Two of the most popular varieties are CCD and CMOS sensors
- Both are arrays of *pixels* → light-sensitive electronic elements which convert incident light energy into charge

Camera characteristics:
- Iris position: regulates amount of light admitted into camera
- Shutter speed: length of integration period
- Gain: amplification of pixel values (noise also amplified)
- Nonlinearity: A pixel can "fill up" with electrons during its integration period (called saturation)
Visual Sensors

- Vision is often considered the most important sense, it is therefore not surprising that visual sensors have advanced so far in recent years.
- Two of the most popular varieties are CCD and CMOS sensors.
- Both are arrays of *pixels* → light-sensitive electronic elements which convert incident light energy into charge; Charge accumulates during the integration period, after which the current charge is “read”.

Camera characteristics:
- Iris position: regulates amount of light admitted into camera.
- Shutter speed: length of integration period.
- Gain: amplification of pixel values (noise also amplified).
- Nonlinearity: A pixel can “fill up” with electrons during its integration period (called saturation).
Vision is often considered the most important sense, it is therefore not surprising that visual sensors have advanced so far in recent years.

Two of the most popular varieties are CCD and CMOS sensors.

Both are arrays of *pixels* → light-sensitive electronic elements which convert incident light energy into charge; Charge accumulates during the integration period, after which the current charge is “read”.

Camera characteristics:

- Iris position: regulates amount of light admitted into camera
- Shutter speed: length of integration period
- Gain: amplification of pixel values (noise also amplified)
- Nonlinearity: A pixel can “fill up” with electrons during the integration period (called saturation)
Visual Sensors

- Vision is often considered the most important sense, it is therefore not surprising that visual sensors have advanced so far in recent years.
- Two of the most popular varieties are CCD and CMOS sensors.
- Both are arrays of pixels → light-sensitive electronic elements which convert incident light energy into charge; Charge accumulates during the integration period, after which the current charge is “read”.
- Camera characteristics:
  - Iris position: regulates amount of light admitted into camera.
Vision is often considered the most important sense, it is therefore not surprising that visual sensors have advanced so far in recent years. Two of the most popular varieties are CCD and CMOS sensors. Both are arrays of pixels → light-sensitive electronic elements which convert incident light energy into charge; Charge accumulates during the integration period, after which the current charge is “read.”

Camera characteristics:
- Iris position: regulates amount of light admitted into camera
- Shutter speed: length of integration period
Vision is often considered the most important sense, it is therefore not surprising that visual sensors have advanced so far in recent years.

Two of the most popular varieties are CCD and CMOS sensors.

Both are arrays of pixels → light-sensitive electronic elements which convert incident light energy into charge; Charge accumulates during the integration period, after which the current charge is “read”.

Camera characteristics:

- Iris position: regulates amount of light admitted into camera
- Shutter speed: length of integration period
- Gain: amplification of pixel values (noise also amplified)
Vision is often considered the most important sense, it is therefore not surprising that visual sensors have advanced so far in recent years.

Two of the most popular varieties are CCD and CMOS sensors.

Both are arrays of *pixels* → light-sensitive electronic elements which convert incident light energy into charge; Charge accumulates during the integration period, after which the current charge is “read.”

Camera characteristics:
- Iris position: regulates amount of light admitted into camera
- Shutter speed: length of integration period
- Gain: amplification of pixel values (noise also amplified)
- Nonlinearity: A pixel can “fill up” with electrons during its integration period (called saturation)
Determining the distance of viewed objects is not as straightforward as it was for active ranging.
Visual Ranging

- Determining the distance of viewed objects is not as straightforward as it was for active ranging.
- 3D world is projected onto 2D image plane; depth information is lost!
Determining the distance of viewed objects is not as straightforward as it was for active ranging.

3D world is projected onto 2D image plane; depth information is lost!

- Recoverable only by positing strong assumptions about the world:
Visual Ranging

- Determining the distance of viewed objects is not as straightforward as it was for active ranging.
- 3D world is projected onto 2D image plane; depth information is lost!
  - Recoverable only by positing strong assumptions about the world:
    - Surface appearance does not significantly vary with viewpoint.
Determining the distance of viewed objects is not as straightforward as it was for active ranging.

3D world is projected onto 2D image plane; depth information is lost!
  - Recoverable only by positing strong assumptions about the world:
    - Surface appearance does not significantly vary with viewpoint
    - Spatial coherence: nearby pixels tend to correspond to objects at the same distance

Stereo Vision:
- We can obtain range information from the left and right images of a pair of cameras if,
  - We know the geometric relationship between the cameras and all of the cameras' properties
  - We know the correspondences between the pixels of the two images (such pixels correspond to the same feature in the world)
Visual Ranging

- Determining the distance of viewed objects is not as straightforward as it was for active ranging.
- 3D world is projected onto 2D image plane; depth information is lost!
  - Recoverable only by positing strong assumptions about the world:
    - Surface appearance does not significantly vary with viewpoint
    - Spatial coherence: nearby pixels tend to correspond to objects at the same distance

Stereo Vision:
We can obtain range information from the left and right images of a pair of cameras if,
- We know the geometric relationship between the cameras and all of the cameras' properties
- We know the correspondences between the pixels of the two images (such pixels correspond to the same feature in the world)
Determining the distance of viewed objects is not as straightforward as it was for active ranging.

3D world is projected onto 2D image plane; depth information is lost!
- Recoverable only by positing strong assumptions about the world:
  - Surface appearance does not significantly vary with viewpoint
  - Spatial coherence: nearby pixels tend to correspond to objects at the same distance

**Stereo Vision:**

We can obtain range information from the left and right images of a pair of cameras if,
Determining the distance of viewed objects is not as straightforward as it was for active ranging.

3D world is projected onto 2D image plane; depth information is lost!
- Recoverable only by positing strong assumptions about the world:
  - Surface appearance does not significantly vary with viewpoint
  - Spatial coherence: nearby pixels tend to correspond to objects at the same distance

**Stereo Vision:**

- We can obtain range information from the left and right images of a pair of cameras if,
  - We know the geometric relationship between the cameras and all of the cameras’ properties
Visual Ranging

- Determining the distance of viewed objects is not as straightforward as it was for active ranging
- 3D world is projected onto 2D image plane; depth information is lost!
  - Recoverable only by positing strong assumptions about the world:
    - Surface appearance does not significantly vary with viewpoint
    - Spatial coherence: nearby pixels tend to correspond to objects at the same distance

Stereo Vision:

- We can obtain range information from the left and right images of a pair of cameras if,
  - We know the geometric relationship between the cameras and all of the cameras’ properties
  - We know the *correspondences* between the pixels of the two images (such pixels correspond to the same feature in the world)
Determining the distance of viewed objects is not as straightforward as it was for active ranging.

3D world is projected onto 2D image plane; depth information is lost!
- Recoverable only by positing strong assumptions about the world:
  - Surface appearance does not significantly vary with viewpoint
  - Spatial coherence: nearby pixels tend to correspond to objects at the same distance

**Stereo Vision:**

- We can obtain range information from the left and right images of a pair of cameras if,
  - We know the geometric relationship between the cameras and all of the cameras’ properties
  - We know the *correspondences* between the pixels of the two images (such pixels correspond to the same feature in the world)
- We consider the second point first, known as the *correspondence problem*
The Correspondence Problem

- We will assume for now that we have a pair of images taken from two identical cameras whose optical axes are parallel and at the same height.
The Correspondence Problem

- We will assume for now that we have a pair of images taken from two identical cameras whose optical axes are parallel and at the same height.
- From such a setup we obtain a pair of images like these,
The Correspondence Problem

- We will assume for now that we have a pair of images taken from two identical cameras whose optical axes are parallel and at the same height.
- From such a setup we obtain a pair of images like these,
The Correspondence Problem

- We will assume for now that we have a pair of images taken from two identical cameras whose optical axes are parallel and at the same height.
- From such a setup we obtain a pair of images like these,
The Correspondence Problem

- We will assume for now that we have a pair of images taken from two identical cameras whose optical axes are parallel and at the same height.

- From such a setup we obtain a pair of images like these,

  ![Images](image1.jpg) ![Images](image2.jpg)

- Note that the shift in image features from one image to another is purely horizontal.
For each pixel \((i,j)\) of the left image, we select the \(M \times M\) window of pixels surrounding it (where \(M\) is odd).
For each pixel \((i,j)\) of the left image, we select the \(M \times M\) window of pixels surrounding it (where \(M\) is odd)

We then search for the best matching window along the corresponding row of the right image.

The horizontal shift \(k\) of the best match is called the disparity. Search must be confined to some fixed search radius. For some pixels no correct match will be available.
Block Matching

- For each pixel \((i, j)\) of the left image, we select the \(M \times M\) window of pixels surrounding it (where \(M\) is odd)
- We then search for the best matching window along the corresponding row of the right image
- We can define “best matching” as the match yielding the lowest Sum of Squared Differences (SSD),

\[
SSD(i, j, k) = \sum_{m=-M}^{M} \sum_{n=-M}^{M} \left[ I_{\text{left}}(i+m, j+n) - I_{\text{right}}(i-k+m, j+n) \right]^2
\]
Block Matching

- For each pixel \((i, j)\) of the left image, we select the \(M \times M\) window of pixels surrounding it (where \(M\) is odd).
- We then search for the best matching window along the corresponding row of the right image.
- We can define “best matching” as the match yielding the lowest Sum of Squared Differences (SSD),

\[
SSD(i, j, k) = \sum_{m=-M/2}^{M/2} \sum_{n=-M/2}^{M/2} [I_{\text{left}}(i+m, j+n) - I_{\text{right}}(i-k+m, j+n)]^2
\]
Block Matching

- For each pixel \((i, j)\) of the left image, we select the \(M \times M\) window of pixels surrounding it (where \(M\) is odd).
- We then search for the best matching window along the corresponding row of the right image.
- We can define “best matching” as the match yielding the lowest Sum of Squared Differences (SSD),

\[
SSD(i, j, k) = \sum_{m=-\frac{M}{2}}^{\frac{M}{2}} \sum_{n=-\frac{M}{2}}^{\frac{M}{2}} \left[ I_{\text{left}}(i + m, j + n) - I_{\text{right}}(i - k + m, j + n) \right]^2
\]
For each pixel \((i, j)\) of the left image, we select the \(M \times M\) window of pixels surrounding it (where \(M\) is odd).

We then search for the best matching window along the corresponding row of the right image.

We can define “best matching” as the match yielding the lowest Sum of Squared Differences (SSD),

\[
SSD(i, j, k) = \sum_{m=-\frac{M}{2}}^{\frac{M}{2}} \sum_{n=-\frac{M}{2}}^{\frac{M}{2}} [I_{\text{left}}(i + m, j + n) - I_{\text{right}}(i - k + m, j + n)]^2
\]

The horizontal shift \(k\) of the best match is called the disparity.
Block Matching

- For each pixel \((i,j)\) of the left image, we select the \(M \times M\) window of pixels surrounding it (where \(M\) is odd).
- We then search for the best matching window along the corresponding row of the right image.
- We can define “best matching” as the match yielding the lowest Sum of Squared Differences (SSD),

\[
SSD(i, j, k) = \sum_{m=-\frac{M}{2}}^{\frac{M}{2}} \sum_{n=-\frac{M}{2}}^{\frac{M}{2}} \left[ I_{\text{left}}(i + m, j + n) - I_{\text{right}}(i - k + m, j + n) \right]^2
\]

- The horizontal shift \(k\) of the best match is called the **disparity**.
- Search must be confined to some fixed search radius.
Block Matching

- For each pixel \((i, j)\) of the left image, we select the \(M \times M\) window of pixels surrounding it (where \(M\) is odd)
- We then search for the best matching window along the corresponding row of the right image
- We can define “best matching” as the match yielding the lowest Sum of Squared Differences (SSD),

\[
SSD(i, j, k) = \sum_{m=-M/2}^{M/2} \sum_{n=-M/2}^{M/2} [I_{left}(i + m, j + n) - I_{right}(i - k + m, j + n)]^2
\]

- The horizontal shift \(k\) of the best match is called the **disparity**
- Search must be confined to some fixed search radius
- For some pixels no correct match will be available
Alternatives to SSD:
Alternatives to SSD:

- Sum of absolute differences

Numerous more sophisticated approaches to stereo correspondence have been proposed [Middlebury Stereo Vision, 2006]
Alternatives to SSD:

- Sum of absolute differences
- Normalized cross-correlation (subtract means from both windows and normalize)
Alternatives to SSD:

- Sum of absolute differences
- Normalized cross-correlation (subtract means from both windows and normalize)

Numerous more sophisticated approaches to stereo correspondence have been proposed [Middlebury Stereo Vision, 2006]
Here is the result of block matching with $7 \times 7$ blocks, matched within a radius of 30.
Here is the result of block matching with $7 \times 7$ blocks, matched within a radius of 30
Here is the result of block matching with $7 \times 7$ blocks, matched within a radius of 30

The computed disparity values ranged from 0 to 30
Here is the result of block matching with $7 \times 7$ blocks, matched within a radius of 30.

The computed disparity values ranged from 0 to 30; These values were mapped to $[0, 255]$ for display.
Here is the result of block matching with $7 \times 7$ blocks, matched within a radius of 30.

The computed disparity values ranged from 0 to 30; these values were mapped to $[0, 255]$ for display; Clearly disparity is proportional to range.
Assume the following:
Assume the following:

- Pixel \((x_l, y_l)\) in the left image corresponds with pixel \((x_r, y_r)\) in the right image (result of correspondence search).
Assume the following:

- Pixel \((x_l, y_l)\) in the left image corresponds with pixel \((x_r, y_r)\) in the right image (result of correspondence search)
- The two cameras are parallel and separated by a distance \(b\), known as the baseline
Assume the following:

- Pixel \((x_l, y_l)\) in the left image corresponds with pixel \((x_r, y_r)\) in the right image (result of correspondence search)
- The two cameras are parallel and separated by a distance \(b\), known as the \textit{baseline}

In the following figure...
Assume the following:
- Pixel \((x_l, y_l)\) in the left image corresponds with pixel \((x_r, y_r)\) in the right image (result of correspondence search).
- The two cameras are parallel and separated by a distance \(b\), known as the baseline.

In the following figure...
- \((x, y, z)\) is the position in space of some object.
Assume the following:
- Pixel \((x_l, y_l)\) in the left image corresponds with pixel \((x_r, y_r)\) in the right image (result of correspondence search).
- The two cameras are parallel and separated by a distance \(b\), known as the baseline.

In the following figure...
- \((x, y, z)\) is the position in space of some object.
- \((x_l, y_l)\) represents the coordinates of the object in the left image plane.
Assume the following:
- Pixel \((x_l, y_l)\) in the left image corresponds with pixel \((x_r, y_r)\) in the right image (result of correspondence search)
- The two cameras are parallel and separated by a distance \(b\), known as the baseline

In the following figure...
- \((x, y, z)\) is the position in space of some object
- \((x_l, y_l)\) represents the coordinates of the object in the left image plane
- \((x_r, y_r)\) represents the coordinates of the object in the right image plane
Assume the following:

- Pixel \((x_l, y_l)\) in the left image corresponds with pixel \((x_r, y_r)\) in the right image (result of correspondence search)
- The two cameras are parallel and separated by a distance \(b\), known as the baseline

In the following figure...

- \((x, y, z)\) is the position in space of some object
- \((x_l, y_l)\) represents the coordinates of the object in the left image plane
- \((x_r, y_r)\) represents the coordinates of the object in the right image plane
- \(f\) is the distance from the lenses to the image plane: the focal length
Obtaining Range by Triangulation

- Assume the following:
  - Pixel \((x_l, y_l)\) in the left image corresponds with pixel \((x_r, y_r)\) in the right image (result of correspondence search)
  - The two cameras are parallel and separated by a distance \(b\), known as the baseline

- In the following figure...
  - \((x, y, z)\) is the position in space of some object
  - \((x_l, y_l)\) represents the coordinates of the object in the left image plane
  - \((x_r, y_r)\) represents the coordinates of the object in the right image plane
  - \(f\) is the distance from the lenses to the image plane: the \textit{focal length}
  - Note: All coordinates are in the world reference frame. There is a further mapping from world to image coordinates.
\[
\frac{x_l}{f}
\]
\[
\frac{x_l}{f} = \frac{-x - b/2}{-z}
\]
\[ \frac{x_l}{f} = \frac{-x - b/2}{-z} = \frac{x + b/2}{z} \]
\[
\frac{x_l}{f} = \frac{-x - b/2}{-z} = \frac{x + b/2}{z} = \frac{x_r}{f}
\]
\[
\frac{x_l}{f} = \frac{-x - b/2}{-z} = \frac{x + b/2}{z} \quad \frac{x_r}{f} = \frac{-x + b/2}{-z}
\]
\[
\frac{x_l}{f} = \frac{-x - b/2}{-z} = \frac{x + b/2}{z} \quad \frac{x_r}{f} = \frac{-x + b/2}{-z} = \frac{x - b/2}{z}
\]
\[
\frac{x_l}{f} = \frac{-x - b/2}{-z} = \frac{x + b/2}{z} \quad \frac{x_r}{f} = \frac{-x + b/2}{-z} = \frac{x - b/2}{z}
\]

\[
\frac{y_l}{f}
\]
\[
\begin{align*}
\frac{x_l}{f} &= \frac{-x - b/2}{-z} = \frac{x + b/2}{z} & \frac{x_r}{f} &= \frac{-x + b/2}{-z} = \frac{x - b/2}{z} \\
\frac{y_l}{f} &= \frac{y_r}{f}
\end{align*}
\]
\[
\frac{x_l}{f} = \frac{-x - b/2}{-z} = \frac{x + b/2}{z} \quad \frac{x_r}{f} = \frac{-x + b/2}{-z} = \frac{x - b/2}{z}
\]

\[
\frac{y_l}{f} = \frac{y_r}{f} = \frac{y}{z}
\]
We have three equations in three unknowns (repeated below)

\[ x_l f = x_r + \frac{b}{2} z \\
\]

\[ x_r f = x_l - \frac{b}{2} z \\
\]

\[ y_l f = y_r f = y z \]

Solve for \((x, y, z)\)
We have three equations in three unknowns (repeated below)

\[
\begin{align*}
    x_l f &= x_r + b / 2 \\
    x_r f &= x_l - b / 2 \\
    y_l f &= y_r
\end{align*}
\]
We have three equations in three unknowns (repeated below)

\[
\frac{x_l}{f}
\]
We have three equations in three unknowns (repeated below)

\[
\frac{x_I}{f} = \frac{x + b/2}{z}
\]
We have three equations in three unknowns (repeated below)

\[
\begin{align*}
\frac{x_l}{f} &= \frac{x + b/2}{z} \\
\frac{x_r}{f} &= \frac{y + b/2}{z}
\end{align*}
\]
We have three equations in three unknowns (repeated below)

\[
\frac{x_l}{f} = \frac{x + b/2}{z} \quad \frac{x_r}{f} = \frac{x - b/2}{z}
\]
We have three equations in three unknowns (repeated below)

\[
\begin{align*}
\frac{x_L}{f} &= \frac{x + b/2}{z} \\
\frac{x_R}{f} &= \frac{x - b/2}{z} \\
\frac{y_L}{f} &= \frac{y}{z}
\end{align*}
\]
We have three equations in three unknowns (repeated below)

\[
\begin{align*}
\frac{x_l}{f} &= \frac{x + b/2}{z} \\
\frac{x_r}{f} &= \frac{x - b/2}{z} \\
\frac{y_l}{f} &= \frac{y_r}{f}
\end{align*}
\]
We have three equations in three unknowns (repeated below)

\[
\frac{x_l}{f} = \frac{x + b/2}{z} \quad \frac{x_r}{f} = \frac{x - b/2}{z}
\]

\[
\frac{y_l}{f} = \frac{y_r}{f} = \frac{y}{z}
\]
We have three equations in three unknowns (repeated below)

\[
\frac{x_l}{f} = \frac{x + b/2}{z} \quad \frac{x_r}{f} = \frac{x - b/2}{z}
\]

\[
\frac{y_l}{f} = \frac{y_r}{f} = \frac{y}{z}
\]

Solve for \((x, y, z)\)
We have three equations in three unknowns (repeated below)

\[
\frac{x_l}{f} = \frac{x + b/2}{z} \quad \frac{x_r}{f} = \frac{x - b/2}{z}
\]

\[
\frac{y_l}{f} = \frac{y_r}{f} = \frac{y}{z}
\]

Solve for \((x, y, z)\)
We have three equations in three unknowns (repeated below)

\[
\frac{x_l}{f} = \frac{x + b/2}{z} \quad \frac{x_r}{f} = \frac{x - b/2}{z} \quad \frac{y_l}{f} = \frac{y_r}{f} = \frac{y}{z}
\]

Solve for \((x, y, z)\)

\[
x = \frac{b(x_l + x_r)}{2(x_l - x_r)}
\]
We have three equations in three unknowns (repeated below)

\[
\frac{x_l}{f} = \frac{x + b/2}{z} \\
\frac{x_r}{f} = \frac{x - b/2}{z}
\]

\[
\frac{y_l}{f} = \frac{y_r}{f} = \frac{y}{z}
\]

Solve for \((x, y, z)\)

\[
x = \frac{b(x_l + x_r)}{2(x_l - x_r)}
\]

\[
y = \frac{b \cdot y_l}{x_l - x_r}
\]
We have three equations in three unknowns (repeated below)

\[
\frac{x_l}{f} = \frac{x + b/2}{z} \quad \frac{x_r}{f} = \frac{x - b/2}{z}
\]

\[
\frac{y_l}{f} = \frac{y_r}{f} = \frac{y}{z}
\]

Solve for \((x, y, z)\)

\[
x = \frac{b(x_l + x_r)}{2(x_l - x_r)}
\]

\[
y = \frac{b \cdot y_l}{x_l - x_r}
\]

\[
z = \frac{b \cdot f}{x_l - x_r}
\]
We have three equations in three unknowns (repeated below)

\[
\frac{x_l}{f} = \frac{x + b/2}{z} \quad \frac{x_r}{f} = \frac{x - b/2}{z}
\]

\[
\frac{y_l}{f} = \frac{y_r}{f} = \frac{y}{z}
\]

Solve for \((x, y, z)\)

\[
x = \frac{b(x_l + x_r)}{2(x_l - x_r)}
\]

\[
y = \frac{b \cdot y_l}{x_l - x_r}
\]

\[
z = \frac{b \cdot f}{x_l - x_r}
\]

The term \(x_l - x_r\) is the **disparity**
\[ z = \frac{b \cdot f}{x_l - x_r} \]

Disparity is large for nearby objects, small for distant objects.

Distance to nearby objects can be measured more accurately.

Disparity is proportional to \( b \).

Increasing \( b \) can increase accuracy, but...

Increasing \( b \) increases the difficulty and expense of the correspondence search (and increases the number of unmatchable pixels).
\[ z = \frac{b \cdot f}{x_l - x_r} \]

\[ \text{disparity} = x_l - x_r = \frac{b \cdot f}{z} \]
\[ z = \frac{b \cdot f}{x_l - x_r} \]

Disparity = \( x_l - x_r = \frac{b \cdot f}{z} \)

- Disparity is large for nearby objects, small for distant objects.
\[ z = \frac{b \cdot f}{x_l - x_r} \]

\[ \text{disparity} = x_l - x_r = \frac{b \cdot f}{z} \]

- Disparity is large for nearby objects, small for distant objects
- Distance to nearby objects can be measured more accurately
\[ z = \frac{b \cdot f}{x_l - x_r} \]

\[
\text{disparity} = x_l - x_r = \frac{b \cdot f}{z} 
\]

- Disparity is large for nearby objects, small for distant objects
  - Distance to nearby objects can be measured more accurately
- Disparity is proportional to \( b \)
\[ z = \frac{b \cdot f}{x_l - x_r} \]

Disparity is large for nearby objects, small for distant objects
- Distance to nearby objects can be measured more accurately
- Disparity is proportional to \( b \)
  - Increasing \( b \) can increase accuracy, but...
\[ z = \frac{b \cdot f}{x_l - x_r} \]

\[
\text{disparity} = x_l - x_r = \frac{b \cdot f}{z}
\]

- Disparity is large for nearby objects, small for distant objects
  - Distance to nearby objects can be measured more accurately
- Disparity is proportional to \( b \)
  - Increasing \( b \) can increase accuracy, but...
  - Increasing \( b \) increases the difficulty and expense of the correspondence search (and increases the number of unmatcheable pixels)
If we don’t have parallel cameras, we can transform our two images such that they appear to have been taken by parallel cameras.
If we don’t have parallel cameras, we can transform our two images such that they appear to have been taken by parallel cameras.
If we don’t have parallel cameras, we can transform our two images such that they appear to have been taken by parallel cameras. This is called **rectification**
If we don’t have parallel cameras, we can transform our two images such that they appear to have been taken by parallel cameras. This is called **rectification**.

The rectified image should generally be made larger than the original to allow for pixels that have shifted outside the original image boundary.
If we don’t have parallel cameras, we can transform our two images such that they appear to have been taken by parallel cameras. This is called **rectification**.

The rectified image should generally be made larger than the original to allow for pixels that have shifted outside the original image boundary.

The final image pixel-to-pixel mapping can be stored in a look-up table to accelerate this process.
The following images were obtained from a Webots robot with two cameras pointed slightly inwards
The following images were obtained from a Webots robot with two cameras pointed slightly inwards.
The following images were obtained from a Webots robot with two cameras pointed slightly inwards.
The following are rectified versions of these images (10% larger than originals)
The following are rectified versions of these images (10% larger than originals)
The following are rectified versions of these images (10% larger than originals)

The top and bottom of the wall are now parallel
Middlebury stereo vision page.