Image Processing

(Year III, 2-nd semester)

Lecture 2: Camera Model

Introduction

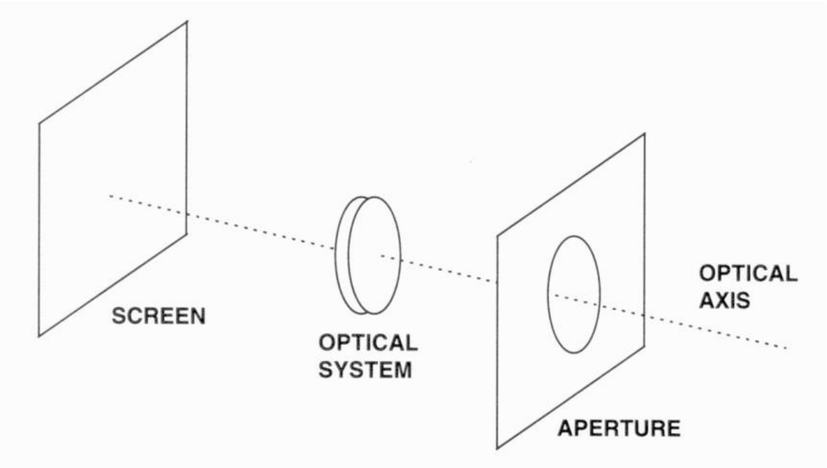
Purpose

Principles of digital image formation

Sensors

Video signal

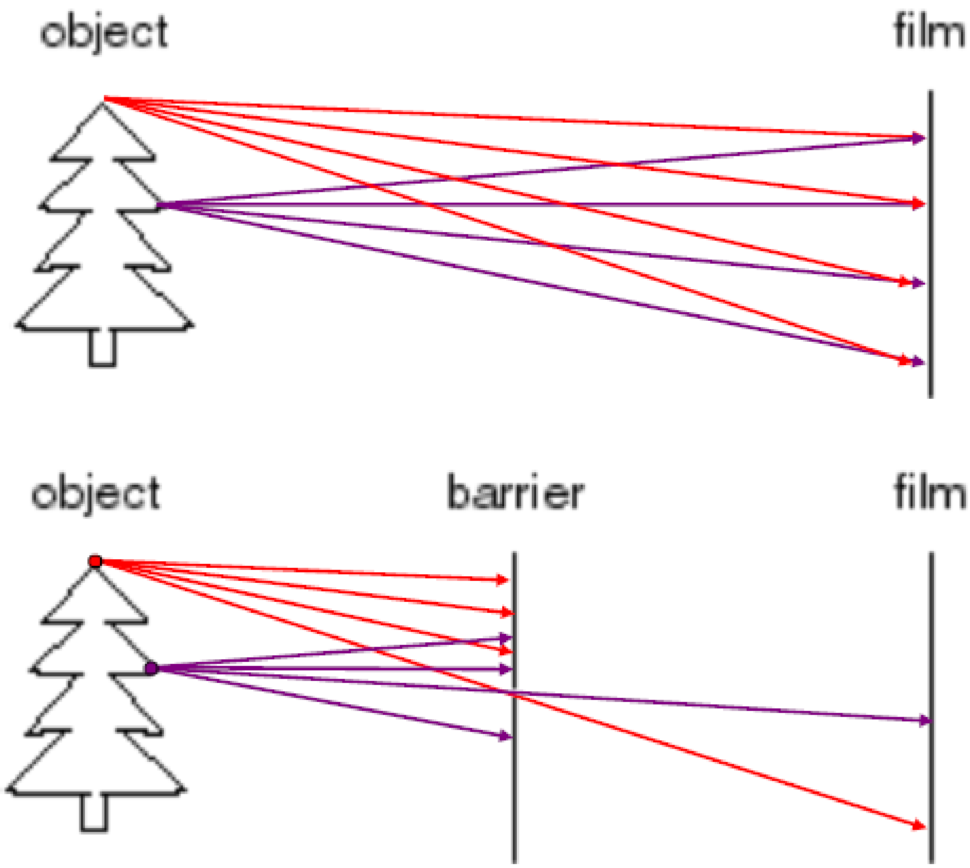
The basic elements of an imaging device



The aperture is a hole or an opening through which light is admitted. It Is implemented through a device, called diaphragm, allowing for different size openings.

Camera model

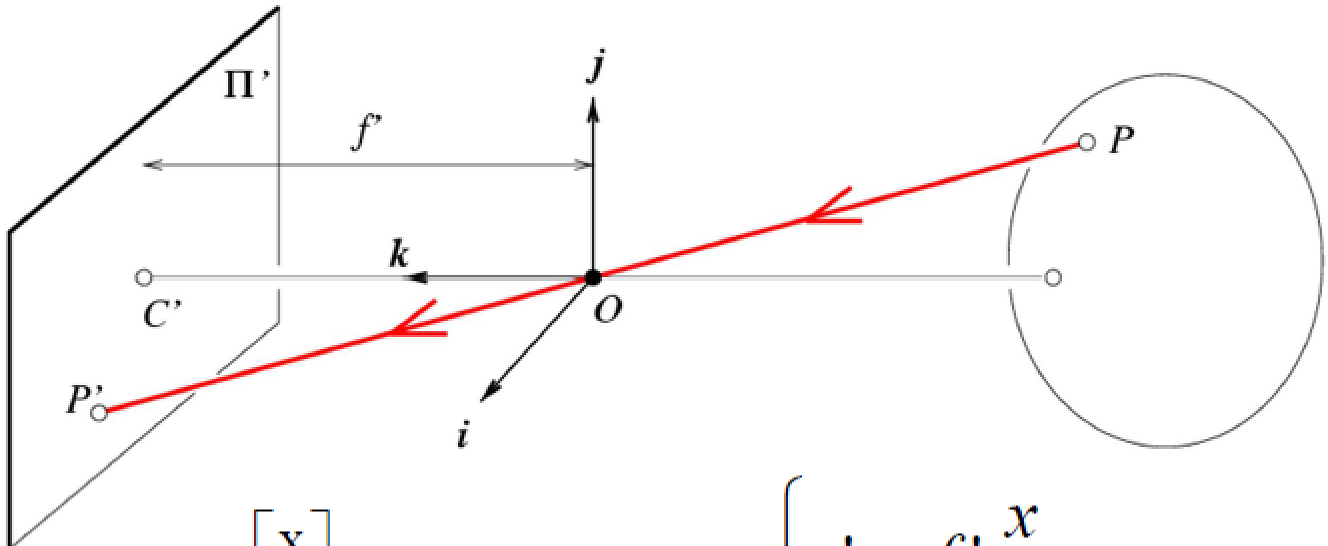
Image formation process



Technical University of Cluj Napoca
Computer Science Department

Camera model

Pinhole or perspective camera model



$$P = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \rightarrow P' = \begin{bmatrix} x' \\ y' \end{bmatrix}$$

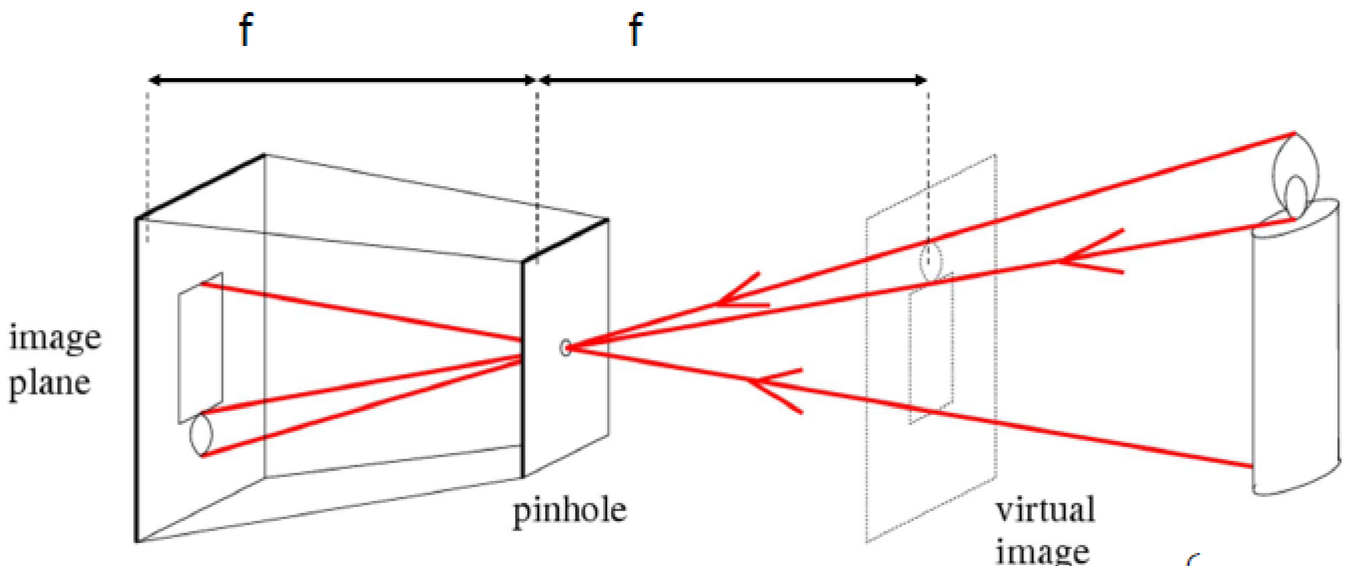
$$\begin{cases} x' = f'\dfrac{x}{z} \\ y' = f'\dfrac{y}{z} \end{cases}$$

Note: z is always negative.

Derived using similar triangles

Camera model

Pinhole or perspective camera model



- Common to draw image plane *in front* of the focal point
- Moving the image plane merely scales the image.

$$\begin{cases} x' = f\,\dfrac{x}{z} \\ y' = f\,\dfrac{y}{z} \end{cases}$$

Camera model

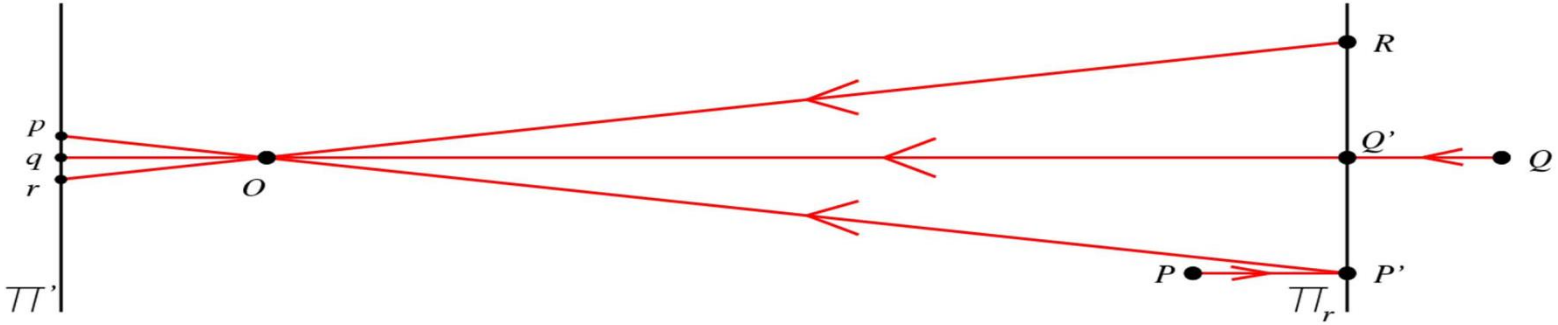
Weak perspective camera model

The relative distance along the optical axis, between two scene points δz is much smaller then the average distance z_{AV} between the camera and these points.

$$\delta z < z_{AV}/20 \qquad (1)$$

The fundamental equation of the weak perspective camera are:

$$x' = f\frac{x}{z} \approx \frac{f}{z_{AV}}x \qquad y' = f\frac{y}{z} \approx \frac{f}{z_{AV}}y \qquad (2)$$



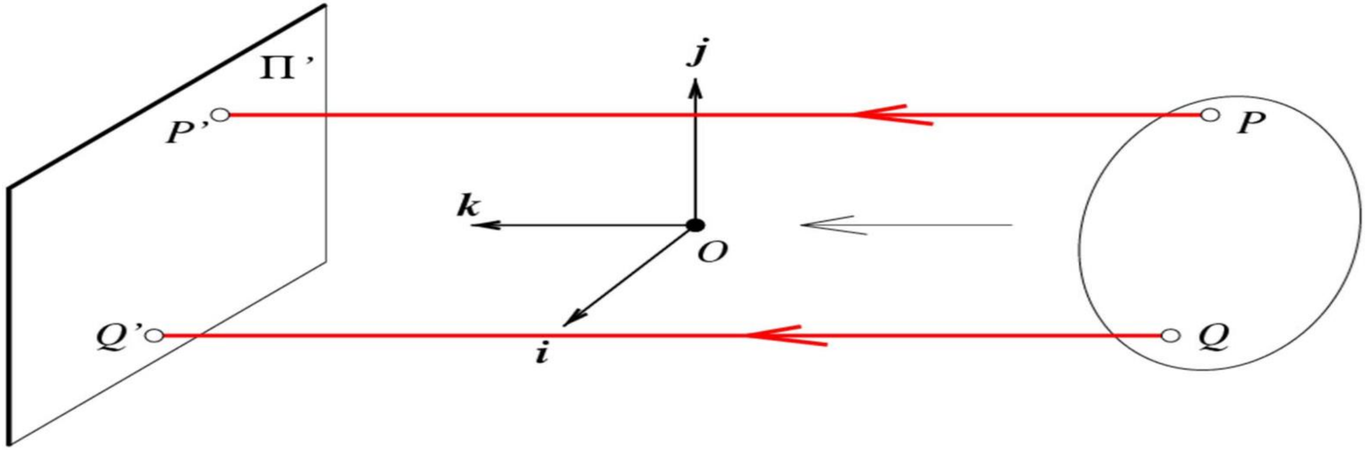
$x' = -m*x$
$y' = -m*y$

$m = -f/z_{AV}$ = magnification

Technical University of Cluj Napoca
Computer Science Department

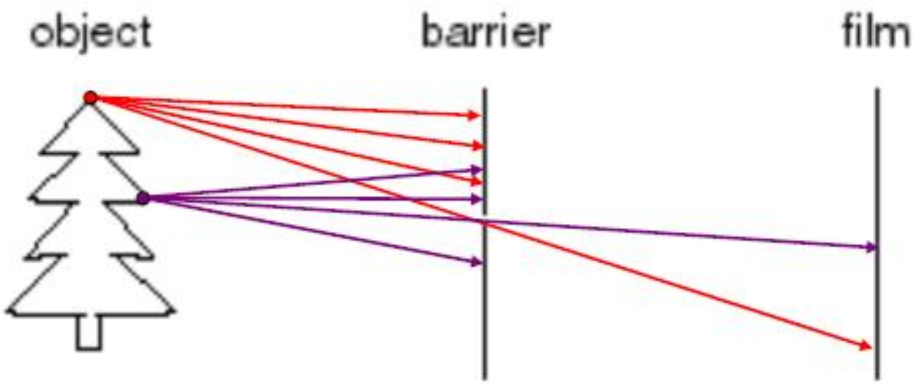
Week perspective camera model

Equations describe a sequence of two transformations:
-**an orthographic projection**, in which points are projected along rays parallel to the optical axis (x'=x; y'=y);
-**an isotropic scaling** by the factor f/z_{AV}



Camera model

Aperture size problem



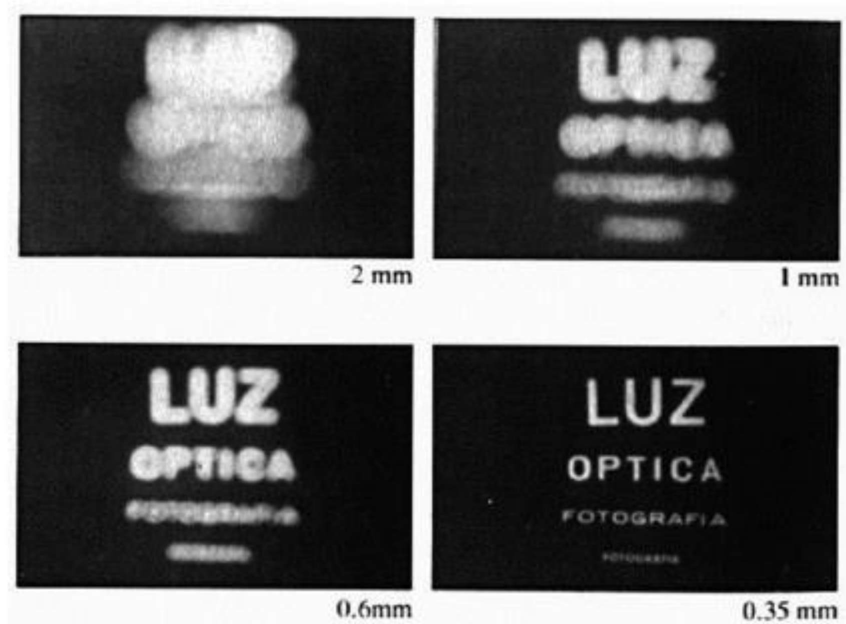
Technical University of Cluj Napoca
Computer Science Department

Camera model

Aperture size problem

Shrinking aperture size

- Rays are mixed up

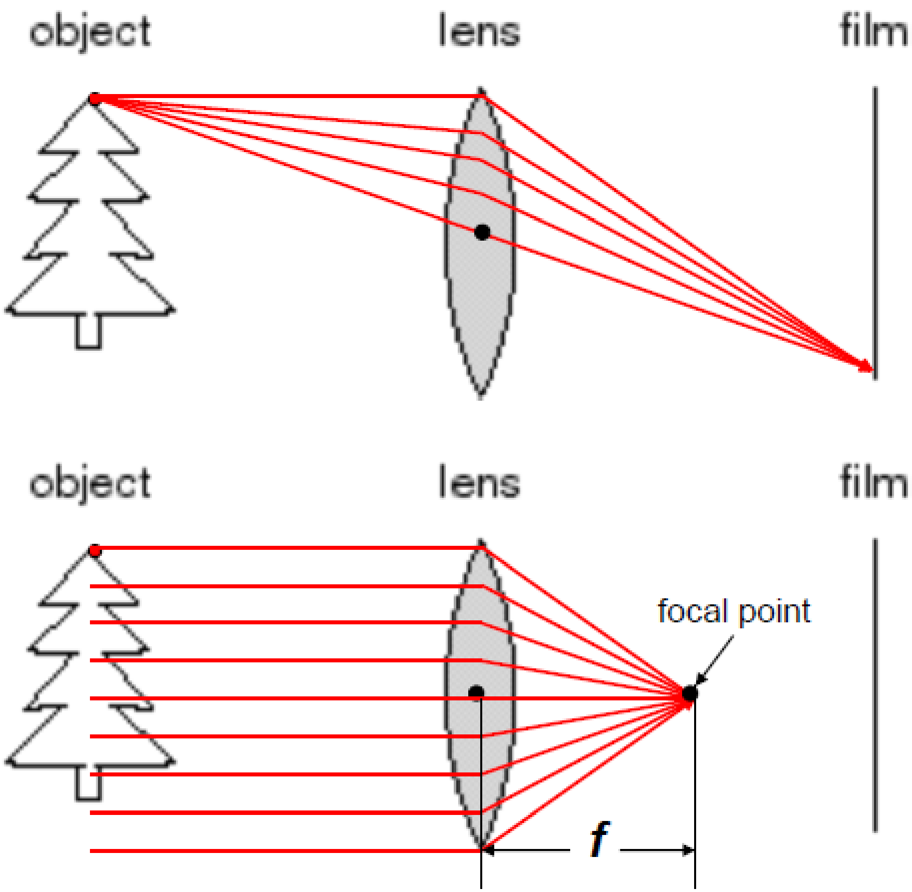


-Why the aperture cannot be too small?
- -Less light passes through
- -Diffraction effect

Adding lenses!

Camera model

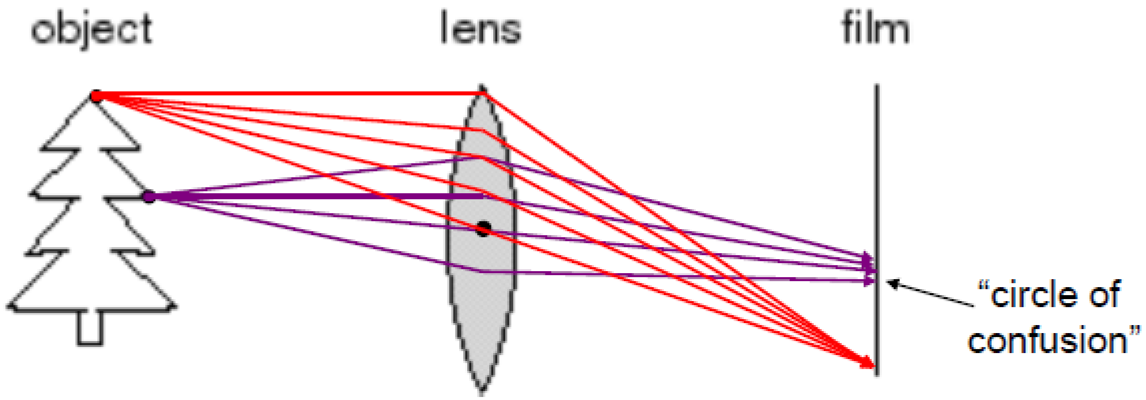
Camera and lenses



Technical University of Cluj Napoca
Computer Science Department

Camera model

Camera and lenses



Camera model

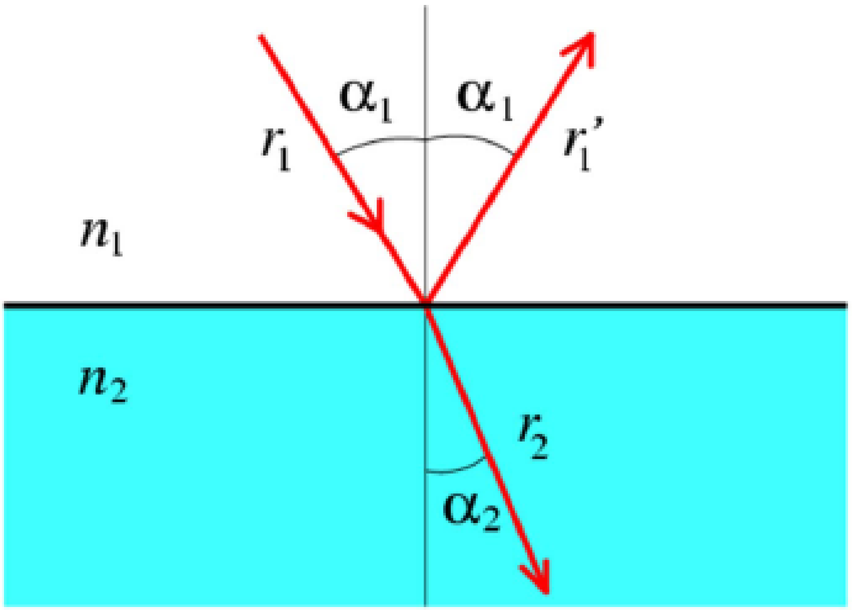
Laws of geometric optics

- Light travels in straight lines in homogeneous medium
- Reflection upon a surface: incoming ray, surface normal, and reflection are co-planar
- Refraction: when a ray passes from one medium to another

Snell's law

$$n_1 \sin \alpha_1 = n_2 \sin \alpha_2$$

α_1 = incident angle
α_2 = refraction angle
n_i = index of refraction



Camera model

The “thin lens” camera model

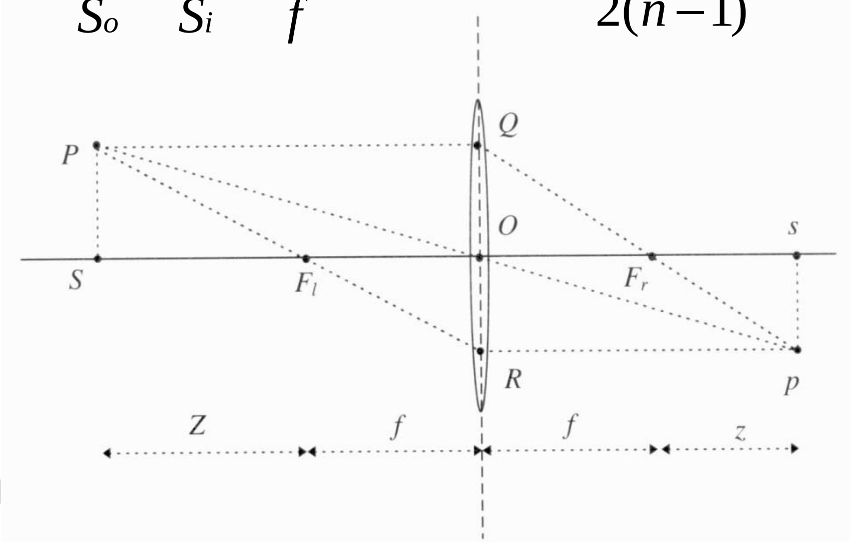
1. **Any ray entering the lens parallel to the optical axis on one side goes through the focus on the other side.**
2. **Any ray entering the lens from the focus on one side emerges parallel to the optical axis on the other size.**
3. **The ray going through the lens center, O, named *principal ray*, goes through point p un-deflected.**

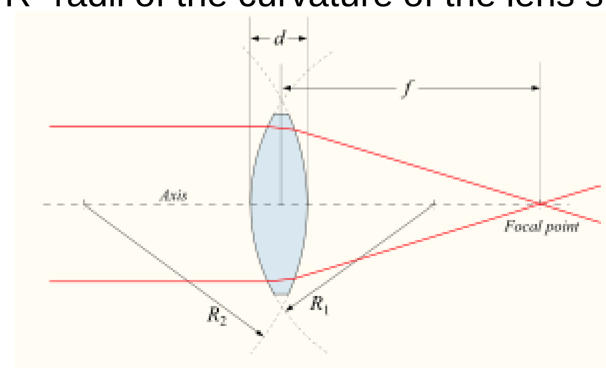
The fundamental equations of thin lenses: **$Z \cdot z = f^2$ (1).** Seting **$S_0 = Z + f$ and $S_i = z + f$** equation (1) can be reduced to the fundamental equations of thin lenses:

$$\frac{1}{S_o} + \frac{1}{S_i} = \frac{1}{f} \quad \textbf{(2)} \qquad f = \frac{R}{2(n-1)} \quad \textbf{(3)}$$

n - index of refraction of the lens material

R- radii of the curvature of the lens surfaces





Positive (converging) lens

A lens can be considered a **thin lens** if $d \ll R_1$ or $d \ll R_2$.

Image focusing

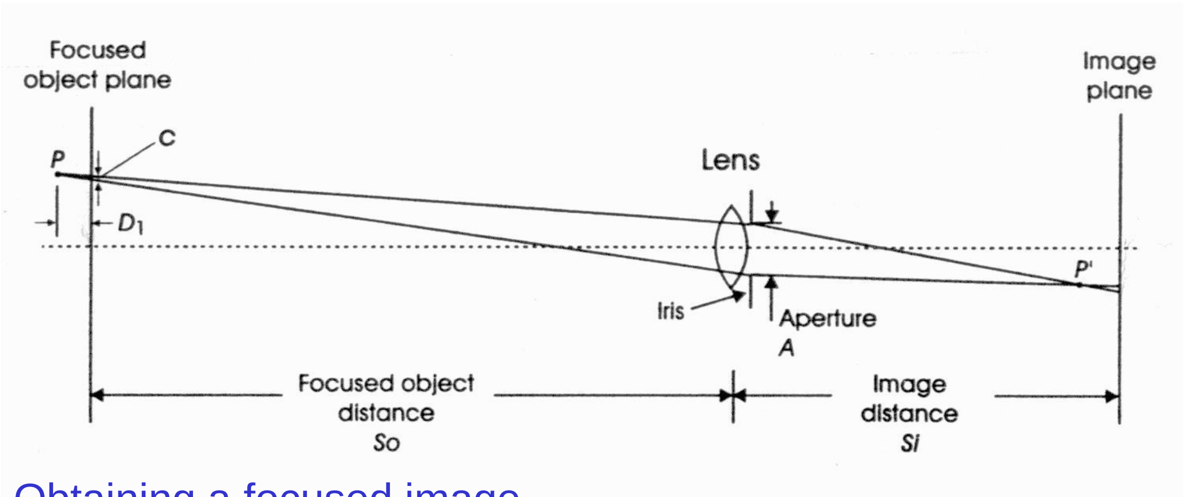


Image is in focus: all rays coming from a single scene point P must converge onto a single point on the image plain p -> sharp image

Image is not in focus: the image of P is spread over a circle -> blurred image

Obtaining a focused image

- pinhole camera/aperture
- optical system (lens)

Measures

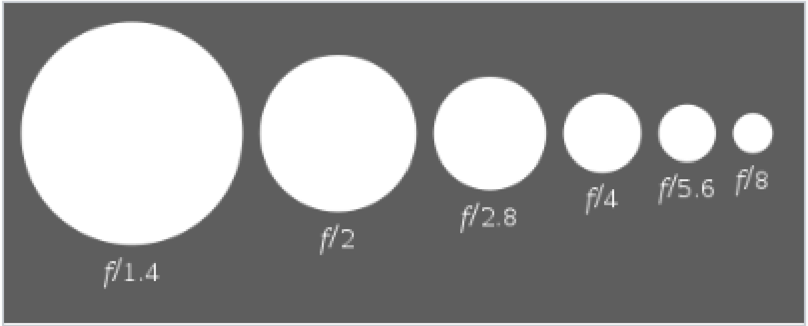
- Circle-of-confusion (c) – its projection on the image plane < 1 pixel (focused image)
- Depth of field – distance (D_1) around the FOP within the diameter of the projection of the circle of confusion(c) on the image remains less than 1 pixel

Camera model

Apertures

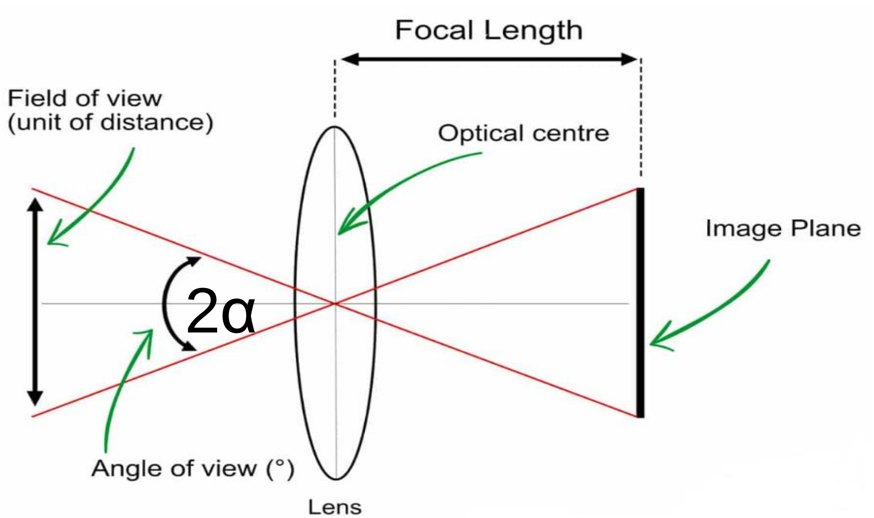
The **f_number** of an optical system is the ratio of the system's focal length to the diameter of the entrance pupil

$$f\_number = \frac{Focal\ Length}{Diameter}$$



Focal Length/ f_number pairs

Field of view (FOV)



$$\text{FOV}=2*\alpha = 2 * \arctan\left(\frac{Imager\ size}{2 * Focal\ Length}\right)$$

Camera model

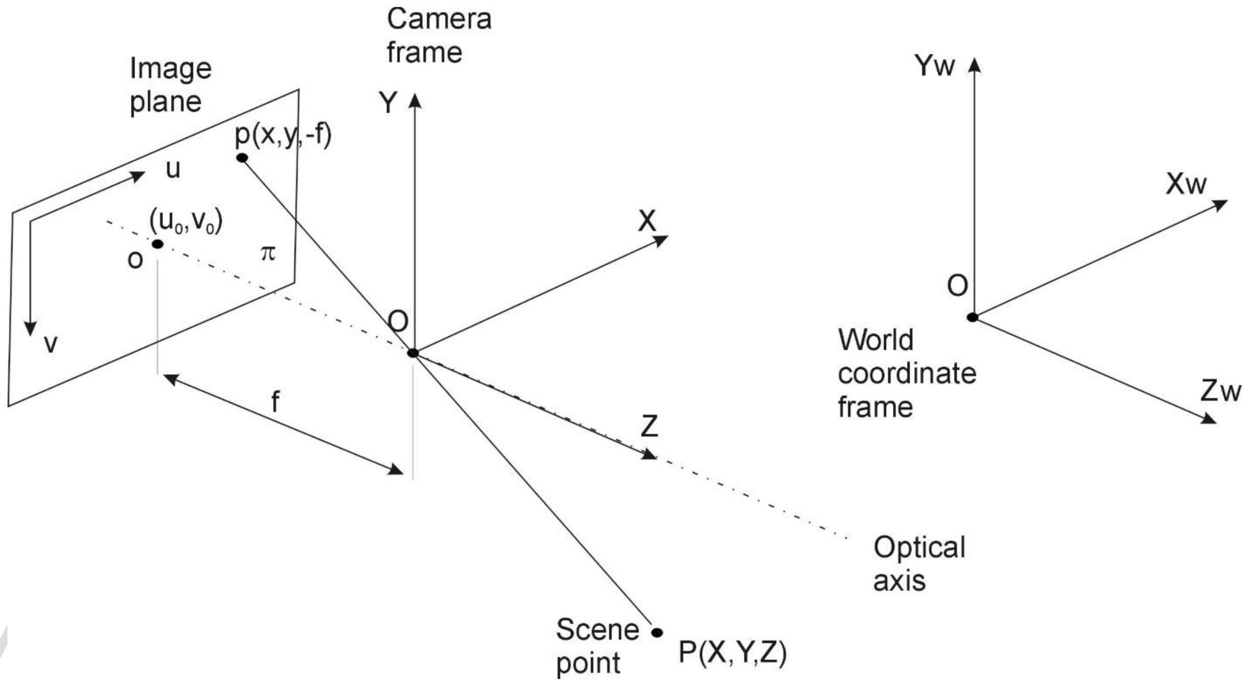
The perspective camera model (pinhole)

- **The most common geometric model of an imaging camera.**
- **Each point in the object space is projected by a straight line through the projection center (pinhole/lens center) into the image plane.**

The fundamental equations of the *perspective camera model* are:

$$\begin{cases} x = f \cdot \dfrac{X_C}{Z_C} \\ y = f \cdot \dfrac{Y_C}{Z_C} \end{cases}$$

***[X_C, Y_C, Z_C]* are the coordinates of point P in the camera coordinate system**



Camera model

Physical camera parameters

Intrinsic parameters := internal camera geometrical and optical characteristics (those that specify the camera itself).

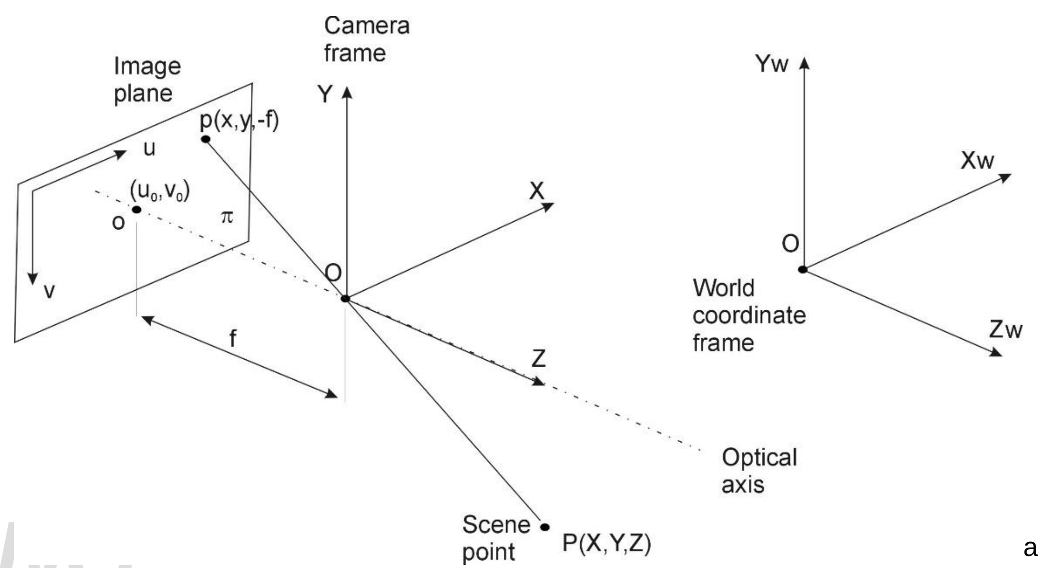
- Focal length := the distance between the optical center of the lens and the image plane: *f* [mm] or [pixels].
- Effective pixel size (dpx, dpy) [mm];
- Principal point := location of the image center in pixel coordinates: (u0,v0)
- Distortion coefficients of the lens: radial (k1, k2) and tangential (p1, p2).

Extrinsic parameters := the 3-D position and orientation of the camera frame relative to a certain world coordinate system:

- Rotation vector r = [Rx, Ry, Rz]^T or

  its equivalent rotation matrix R
- Translation vector T = [Tx, Ty, Tz]^T ;

In multi-camera (stereo) systems, the extrinsic parameters also describe the relationship between the cameras



Camera model

Physical camera parameters

Intrinsic parameters := internal camera geometrical and optical characteristics (those that specify the camera itself).

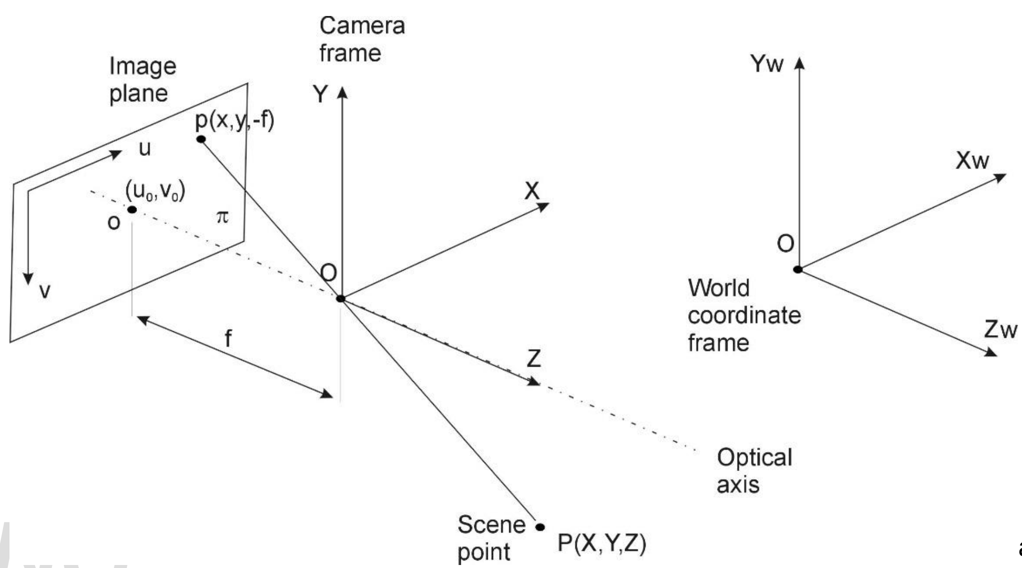
- Focal length := the distance between the optical center of the lens and the image plane: *f* [mm] or [pixels].
- Effective pixel size (dpx, dpy) [mm];
- Principal point := location of the image center in pixel coordinates: (u0,v0)
- Distortion coefficients of the lens: radial (k1, k2) and tangential (p1, p2).

Extrinsic parameters := the 3-D position and orientation of the camera frame relative to a certain world coordinate system:

- Rotation vector r = [Rx, Ry, Rz]^T or

  its equivalent rotation matrix R
- Translation vector T = [Tx, Ty, Tz]^T ;

In multi-camera (stereo) systems, the extrinsic parameters also describe the relationship between the cameras



Camera frame ↔ image plane transformation

Camera frame ⇒ image plane transformation

(*projection / normalization*) : P = $[X_C, Y_C, Z_C]^T$ [metric units] ⇒ p = $[u, v]^T$ [pixels]

1. Transform P = $[X_C, Y_C, Z_C]^T$ ⇒ p = $[x, y, -f]^T$

Fundamental equations of the *perspective camera model* normalized with *1/Z*:

$$\begin{bmatrix} x \\ y \end{bmatrix} = f \begin{bmatrix} X_c / Z_c \\ Y_c / Z_c \end{bmatrix} = f \begin{bmatrix} x_N \\ y_N \end{bmatrix}$$

f – focal distance [metric units]

2. Transform **p** $[x, y]^T$ [metric units] ⇒ image coordinates $[u, v]^T$ [pixels]

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} D_u \cdot x \\ D_v \cdot y \end{bmatrix} + \begin{bmatrix} u_0 \\ v_0 \end{bmatrix}$$

Du, Dv - coefficients needed to transform metric units to pixels: Du = 1 / dpx; Dv = 1 / dpy

1 + 2 ⇒ projection equation: $$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = A \cdot \begin{bmatrix} x_N \\ y_N \\ 1 \end{bmatrix}$$

$$A = \begin{bmatrix} f_X & 0 & u_0 \\ 0 & f_Y & v_0 \\ 0 & 0 & 1 \end{bmatrix}$$

A – is the camera matrix:

f_X – is the focal distance expressed in units of horizontal pixels: $$f_X = f \cdot D_u = \frac{f}{dpx}$$

f_Y – is the focal distance expressed in units of vertical pixels: $$f_Y = f \cdot D_v = \frac{f}{dpy}$$

Camera frame ↔ image plane transformation

Image plane transformation ⇒ camera frame

(*reconstruction*) : p = [u, v]^T [pixels] ⇒ P = [X_C, Y_C, Z_C]^T [metric units]

$$\begin{bmatrix} x_N \\ y_N \\ 1 \end{bmatrix} = A^{-1} \cdot \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}$$

Notes:

With one camera we cannot measure depth (Z). We can determine only the projection equation / normalized coordinates:

$$\begin{bmatrix} x_N \\ y_N \end{bmatrix} = \begin{bmatrix} X_C / Z_C \\ Yc / Z_C \end{bmatrix}$$

To measure the depth (Z) a stereo system (2 cameras) is needed

Technical University of Cluj Napoca
Computer Science Department

Camera model

Modeling the lens distortions

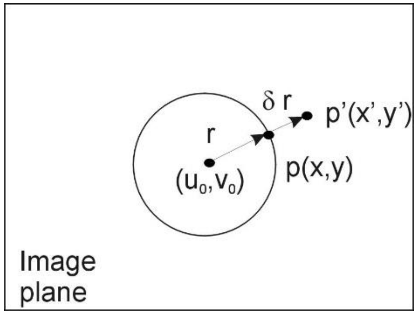
Radial lens distortion

Causes the actual image point to be displaced radially in the image plane

$$\begin{bmatrix} \partial x^r \\ \partial y^r \end{bmatrix} = \begin{bmatrix} x \cdot \left(k_1 \cdot r^2 + k_2 \cdot r^4 + \ldots\right) \\ y \cdot \left(k_1 \cdot r^2 + k_2 \cdot r^4 + \ldots\right) \end{bmatrix}$$

$r^2 = x^2 + y^2$;

$k_1, k_2, \ldots$ - radial distortion coefficients



Tangential distortion

Appears if the centers of curvature of the lenses' surfaces are not strictly collinear

$$\begin{bmatrix} \partial x^t \\ \partial y^t \end{bmatrix} = \begin{bmatrix} 2p_1 \cdot xy + p_2(r^2 + 2x^2) \\ p_1(r^2 + 2y^2) + 2p_2 \cdot xy \end{bmatrix}$$

p_1, p_2 – tangential distortion coefficients

Transform **p** [x, y]T [metric units] ⇒ image coordinates [u, v]T [pixels]:

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} D_u \cdot (x + \partial x^r + \partial x^t) \\ D_v \cdot (y + \partial y^r + \partial y^t) \end{bmatrix} + \begin{bmatrix} u_0 \\ v_0 \end{bmatrix}$$

⇒The projection equations become non-linear

Solution: perform distortion correction on image and afterwards linear projection

Camera model

Distortion correction algorithm

Evaluate the distortion coefficients: k_1, k_2, p_1, p_2

Use calibration tools: http://www.vision.caltech.edu/bouguetj/calib_doc/

- Camera Calibration Toolbox for Matlab

The principle behind the distortion correction algorithm is **the existence of a bivalent correspondence between the distorted image pixels (x', y') and the undistorted image pixels (x, y)**

The correction algorithm

For each pixel (u, v) from the corrected image D:

- compute the (x, y) coordinates (1)
- compute the (x', y') coordinates in the distorted image S (2)
- compute the (u', v') coordinates in the distorted image S (3)
- *D(u, v)=S(u', v')*

Camera model

Distortion correction algorithm

$$\begin{cases} x = \dfrac{u - u_0}{f_x} \\ y = \dfrac{v - v_0}{f_y} \end{cases} \qquad \textbf{(1)}$$

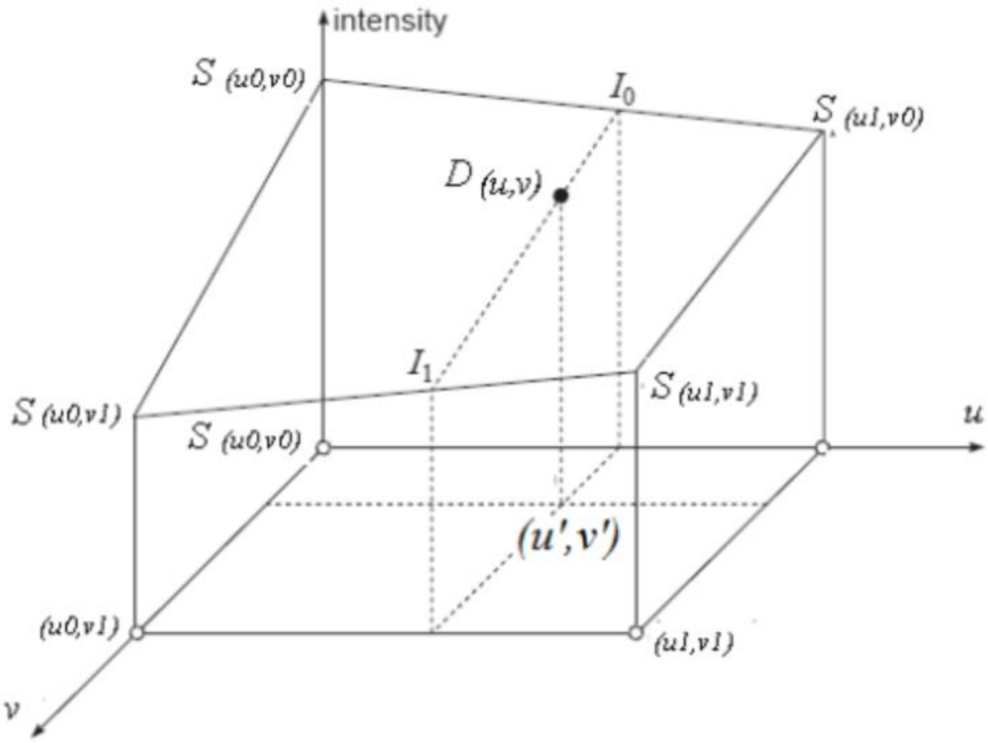
$$(x', y') = (x + \delta x, y + \delta y) \qquad (2)$$

$$\begin{cases} u' = u_0 + x' f_x \\ v' = v_0 + y' f_y \end{cases} \qquad (3)$$

Technical University of Cluj Napoca
Computer Science Department

Camera model

Distortion correction algorithm



$u_0=int(u');$
$v_0=int(v');$

$u_1=u_0+1;$
$v_1=v_0+1;$

$I_0=S(u_0,v_0)\cdot(u_1-u')+S(u_1,v_0)\cdot(u'-v_0);$
$I_1=S(u_0,v_1)\cdot(u_1-u')+S(u_1,v_1)\cdot(u'-v_0);$

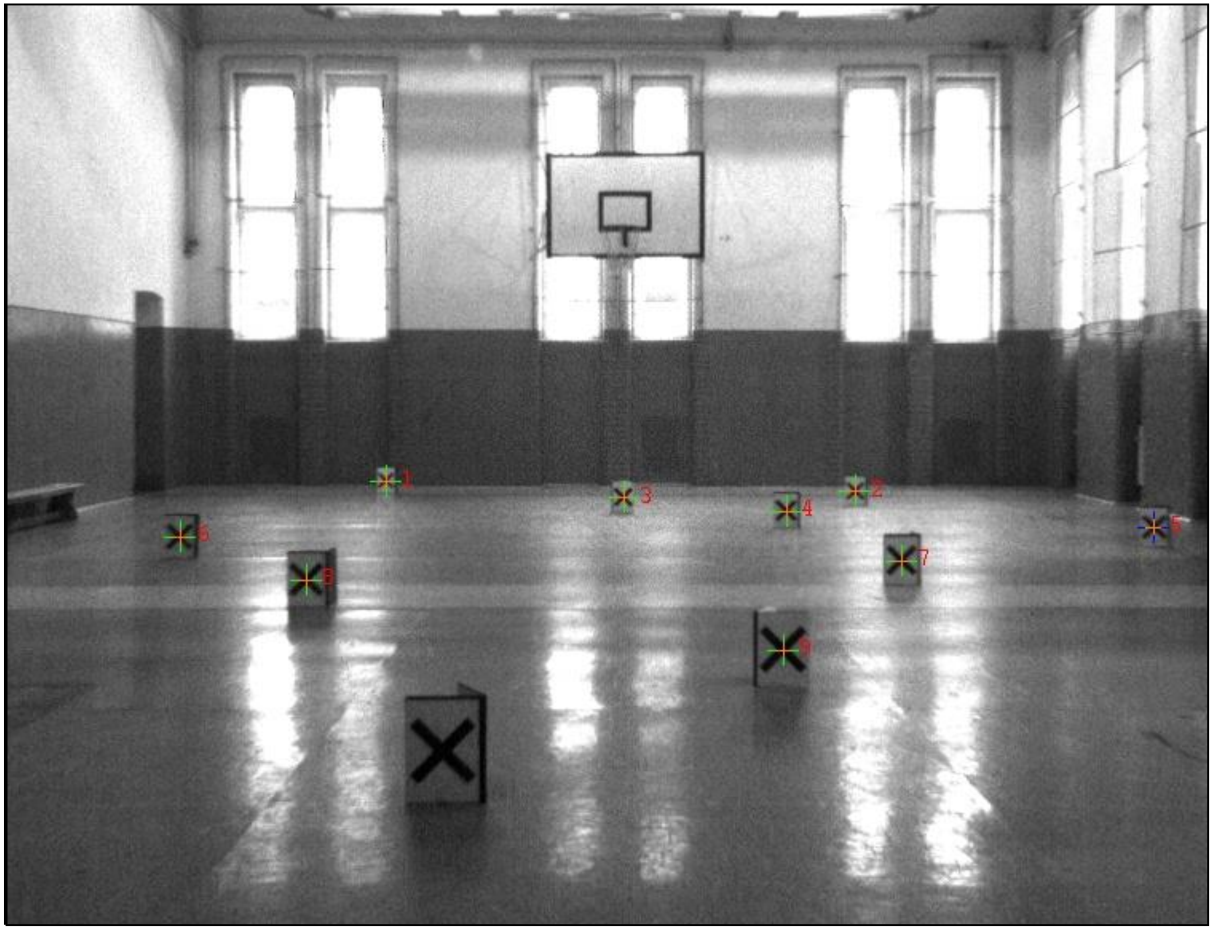
$D(u,v)=I_0\cdot(v_1-v')+I_1\cdot(v'-v_0);$

Bilinear interpolation of the destination pixel Intensity *D(u, v)* starting from the floating point coordinates of the source pixel *(u', v')*

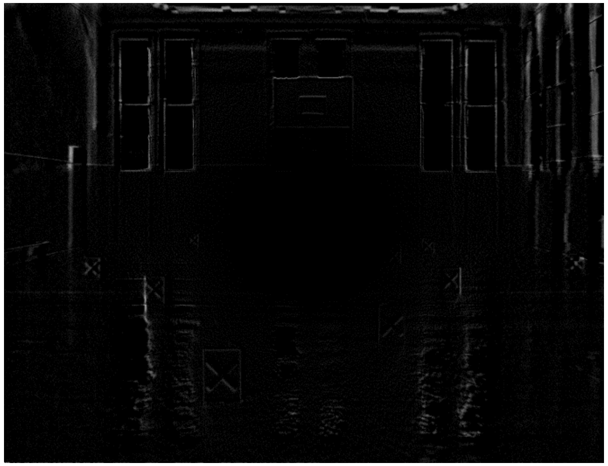
Technical University of Cluj Napoca
Computer Science Department

Lenses distortion correction

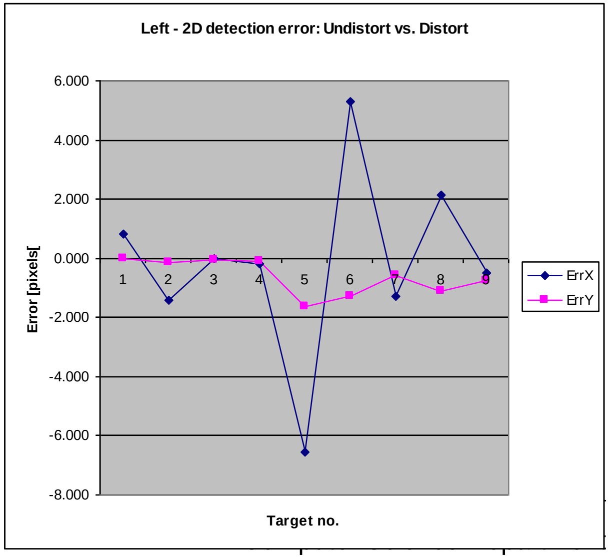
8.5 mm lens, CCD camera



Undistorted image

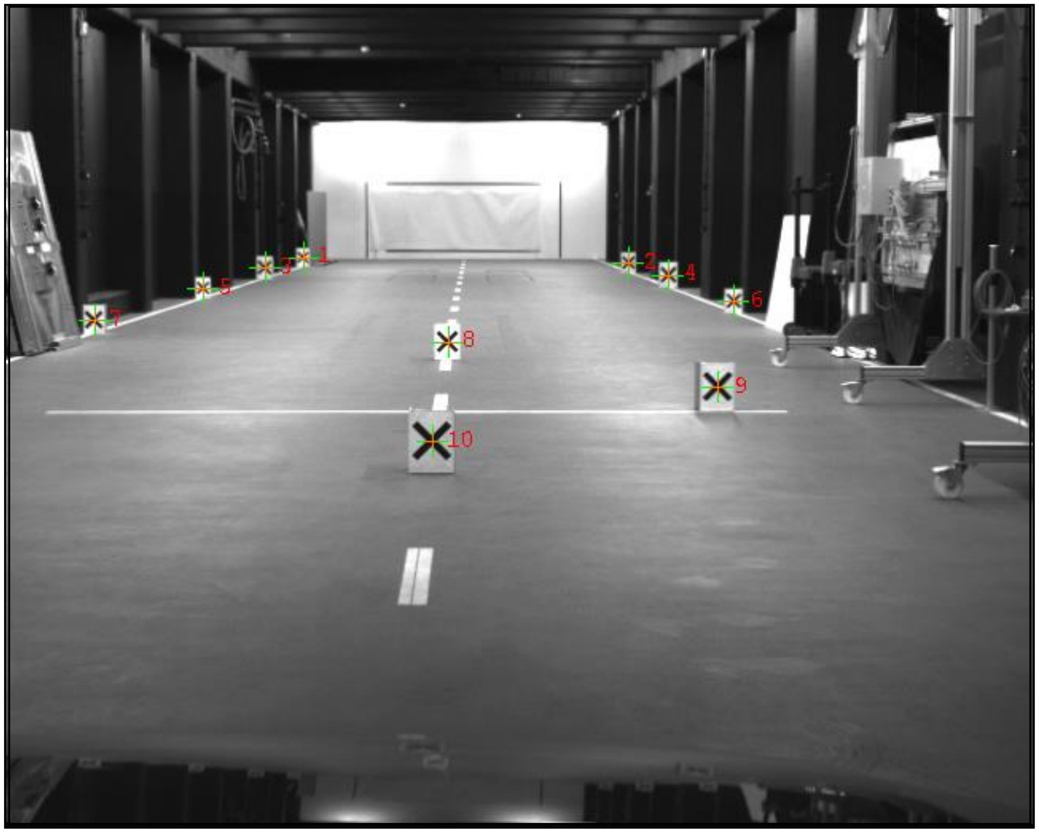


Difference image

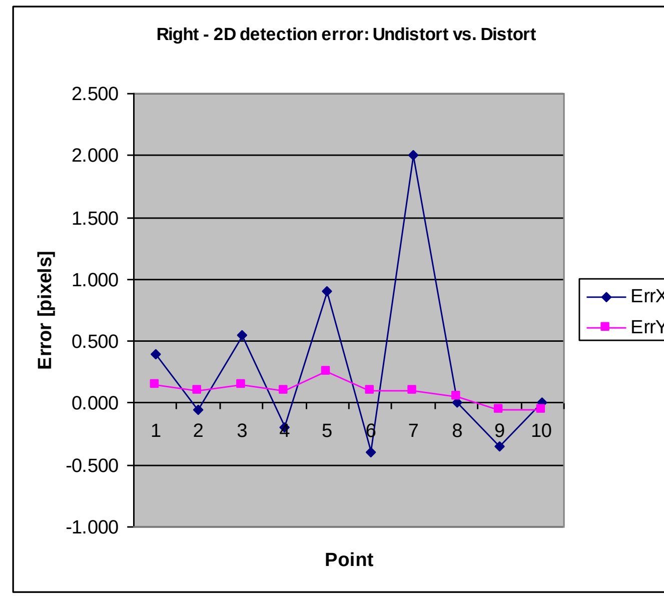


Lenses distortion correction

16 mm lens, CCD camera



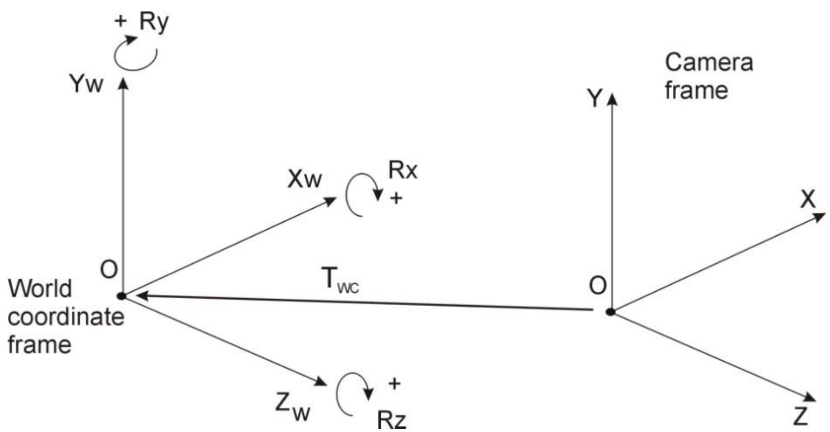
Undistorted image



Camera frame ↔ world reference frame transformation

Direct mapping (world ⇒ camera)

$\mathbf{XX}_W = [\, X_W , Y_W , Z_W \,]^T$ (world coordinate system - WRF) $\Rightarrow \mathbf{XX}_C = [X_C , Y_C , Z_C]^T$ (camera coordinate system – CRF)



$$\mathbf{XX}_C = \mathbf{R}_{WC} \cdot \mathbf{XX}_W + \mathbf{T}_{WC}$$

where:

$\mathbf{T}_{WC} = [\, Tx, Ty, Tz \,]^T$ – world to camera translation vector;
$\mathbf{R}_{WC}$ – world to camera rotation matrix:

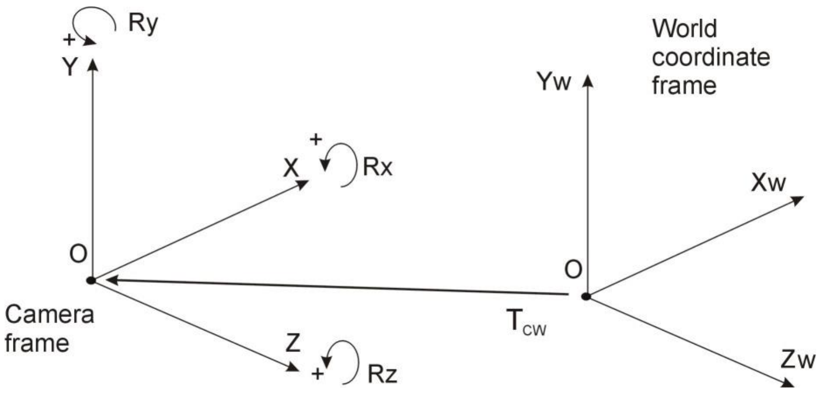
Technical University of Cluj Napoca
Computer Science Department

Camera frame ↔ world reference frame transformation

Inverse mapping (camera ⇒ world)

$\mathbf{XX}_C = [X_C\ ,\ Y_C\ ,\ Z_C]^T$ (camera coordinate system – CRF) ⇒ $\mathbf{XX}_W = [\ X_W\ ,\ Y_W\ ,\ Z_W\ ]^T$ (world coordinate system - WRF)

$$\mathbf{XX}_W = \mathbf{R}_{WC}{}^{-1} \cdot (\mathbf{XX}_C - \mathbf{T}_{WC})$$



Rotation matrix is orthogonal [Trucco1998]:

$$\mathbf{R} \cdot \mathbf{R}^T = \mathbf{R}^T \cdot \mathbf{R} = 1 \Rightarrow \mathbf{R}^T = \mathbf{R}^{-1}$$

$$\mathbf{XX}_W = \mathbf{R}_{WC}{}^T \cdot (\mathbf{XX}_C - \mathbf{T}_{WC}) = \mathbf{R}_{CW} \cdot (\mathbf{XX}_C + \mathbf{T}_{CW})$$

where:

$\mathbf{T}_{CW} = [T_X\ T_Y\ T_Z]^T$ – camera to world translation vector $\quad T_{CW} = -T_{WC}$

$\mathbf{R}_{CW}$ – camera to world translation vector $\quad R_{CW} = R_{WC}{}^T$

Rotation Matrix

World-to-camera

$$\mathbf{R}_{WC} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} = \begin{bmatrix} \mathbf{n}^{XW} & \mathbf{n}^{YW} & \mathbf{n}^{ZW} \end{bmatrix} = \begin{bmatrix} n_X^{XW} & n_X^{YW} & n_X^{ZW} \\ n_Y^{XW} & n_Y^{YW} & n_Y^{YW} \\ n_Z^{XW} & n_Z^{YW} & n_Z^{YW} \end{bmatrix}$$

$\mathbf{n}^{XW} = \begin{bmatrix} n_X^{XW} & n_Y^{XW} & n_Z^{XW} \end{bmatrix}^T$ – normal vector of **OX**$_W$ axis in the CRF

$\mathbf{n}^{YW} = \begin{bmatrix} n_X^{YW} & n_Y^{YW} & n_Z^{YW} \end{bmatrix}^T$ – normal vector of **OY**$_W$ axis in the CRF

$\mathbf{n}^{ZW} = \begin{bmatrix} n_X^{ZW} & n_Y^{ZW} & n_Z^{ZW} \end{bmatrix}^T$ – normal vector of **OZ**$_W$ axis in the CRF

Camera-to-world

$$\mathbf{R}_{CW} = \mathbf{R}_{WC}^{\ T} = \begin{bmatrix} r_{11} & r_{21} & r_{31} \\ r_{12} & r_{22} & r_{32} \\ r_{13} & r_{23} & r_{33} \end{bmatrix} = \begin{bmatrix} \mathbf{n}^{XC} & \mathbf{n}^{YC} & \mathbf{n}^{ZC} \end{bmatrix} = \begin{bmatrix} n_X^{XC} & n_X^{YC} & n_X^{ZC} \\ n_Y^{XC} & n_Y^{YC} & n_Y^{YC} \\ n_Z^{XC} & n_Z^{YC} & n_Z^{YC} \end{bmatrix}$$

$\mathbf{n}^{XC} = \begin{bmatrix} n_X^{XC} & n_Y^{XC} & n_Z^{XC} \end{bmatrix}^T$ – normal vector of **OX**$_C$ axis in the WRF

$\mathbf{n}^{YC} = \begin{bmatrix} n_X^{YC} & n_Y^{YC} & n_Z^{YC} \end{bmatrix}^{YT}$ – normal vector of **OY**$_C$ axis in the WRF

$\mathbf{n}^{ZC} = \begin{bmatrix} n_X^{ZC} & n_Y^{ZC} & n_Z^{ZC} \end{bmatrix}^T$ – normal vector of **OZ**$_C$ axis in the WRF

Rotation Matrix ↔ Rotation Vector

Rotation vector – Rotation matrix

$$\mathbf{r} = [\theta, \psi, \gamma]^T \qquad \mathbf{R} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}$$

$$\mathbf{R}x = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{pmatrix} \quad \mathbf{R}y = \begin{pmatrix} \cos\psi & 0 & \sin\psi \\ 0 & 1 & 0 \\ \sin\psi & 0 & \cos\psi \end{pmatrix} \quad \mathbf{R}z = \begin{pmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$\mathbf{R} = \mathbf{R}x\mathbf{R}y\mathbf{R}z$$

Technical University of Cluj Napoca
Computer Science Department

Rotation Matrix ↔ Rotation Vector

Rotation vector

$\mathbf{r}_{WC} = [R_X\ R_Y\ R_Z]^T$ (R_X - pitch, R_Y - yaw, R_Z - tilt / roll)

$r_{WC} \Rightarrow R_{WC}$ transform:

$$
\begin{aligned}
r_{11} &= \cos(R_Y)\cos(R_Z) \\
r_{12} &= \sin(R_X)\sin(R_Y)\cos(R_Z) - \cos(R_X)\sin(R_Z) \\
r_{13} &= \cos(R_X)\sin(R_Y)\cos(R_Z) + \sin(R_X)\sin(R_Z) \\
r_{21} &= \cos(R_Y)\sin(R_Z) \\
r_{22} &= \sin(R_X)\sin(R_Y)\sin(R_Z) + \cos(R_X)\cos(R_Z) \\
r_{23} &= \cos(R_X)\sin(R_Y)\sin(R_Z) - \sin(R_X)\cos(R_Z) \\
r_{31} &= -\sin(R_Y) \\
r_{32} &= \sin(R_X)\cos(R_Y) \\
r_{33} &= \cos(R_X)\cos(R_Y)
\end{aligned}
$$

$R_{WC} \Rightarrow r_{WC}$ transform:

$$R_Y = \arcsin(r_{31})$$

If $\cos(R_Y) \neq 0$:

$$R_X = \operatorname{atan2}\left(-\frac{r_{32}}{\cos(R_Y)}, \frac{r_{33}}{\cos(R_Y)}\right)$$

$$R_Z = -\operatorname{atan2}\left(-\frac{r_{21}}{\cos(R_Y)}, \frac{r_{11}}{\cos(R_Y)}\right)$$

If $\cos(R_Y) = 0$:

$$R_X = \operatorname{atan2}(r_{12}, r_{22})$$

$$R_Z = 0$$

Technical University of Cluj Napoca
Computer Science Department

3D (world) ⇒ 2D (image) mapping using the Projection Matrix

Projection matrix

$$\mathbf{P} = \mathbf{A} \cdot \left[\mathbf{R}_{WC} \mid \mathbf{T}_{WC}\right]$$

The projection equation of a 3D world point [X_W , Y_W , Z_W] expressed in normalized coordinates :

$$s \cdot \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} x_S \\ y_S \\ z_S \end{bmatrix} = \mathbf{P} \cdot \begin{bmatrix} X_W \\ Y_W \\ Z_W \\ 1 \end{bmatrix}$$

$s=z_S$ – scaling factor

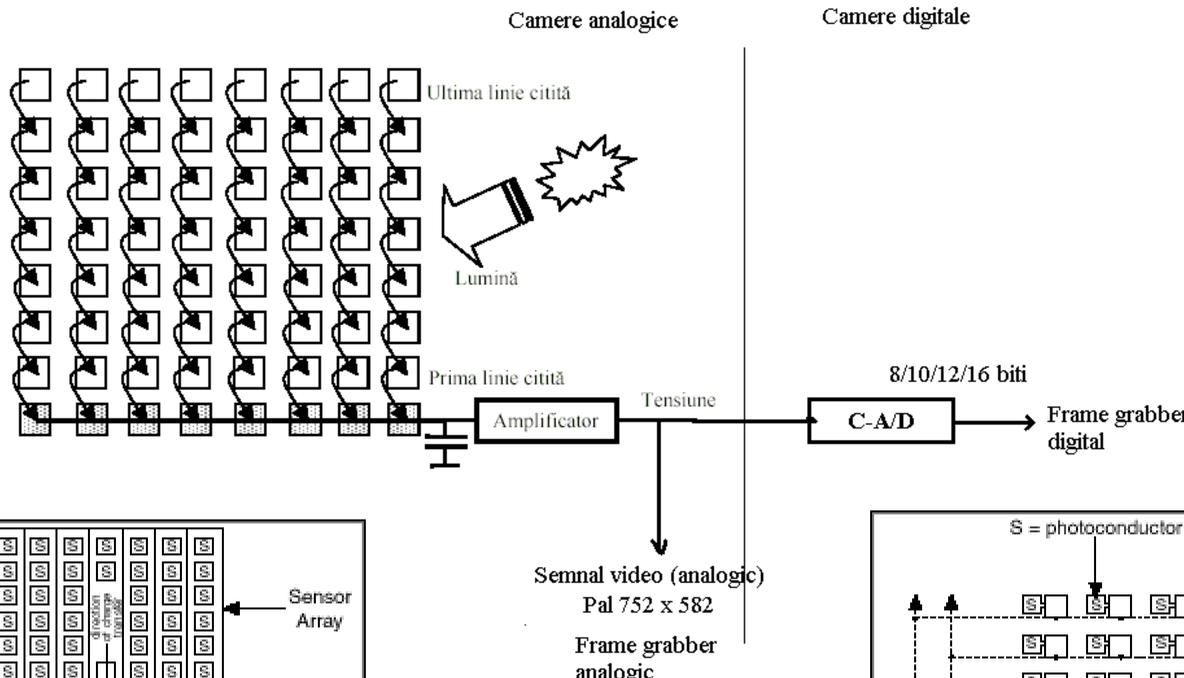
Obtaining the 2D image coordinates from normalized coordinate

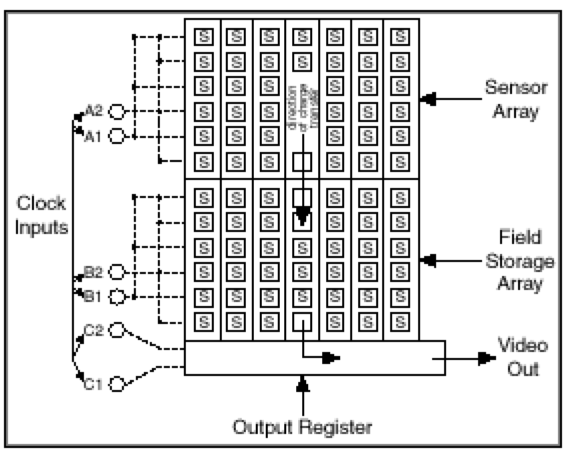
$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} x_S / z_S \\ y_S / z_S \end{bmatrix}$$

Imaging sensors

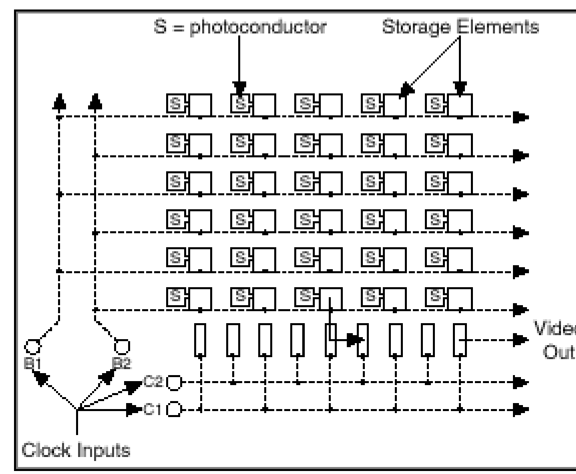
Sensor types

CCD (Charged Coupled Device)





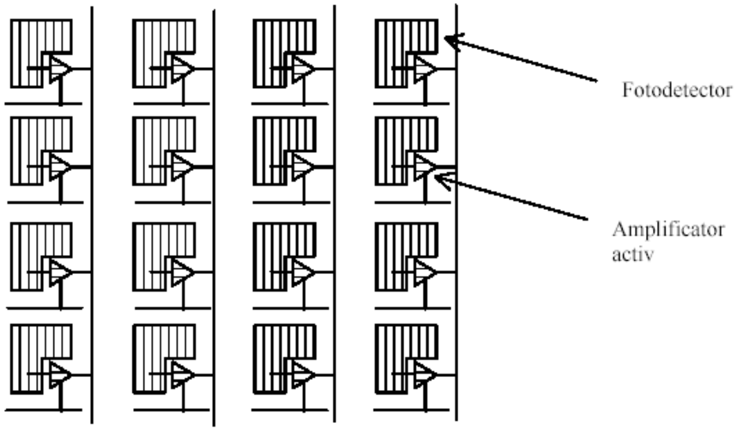
Frame Transfer Architecture



Interline Transfer Architecture

Sensor types

CMOS



Imaging sensors

CMOS vs. CCD

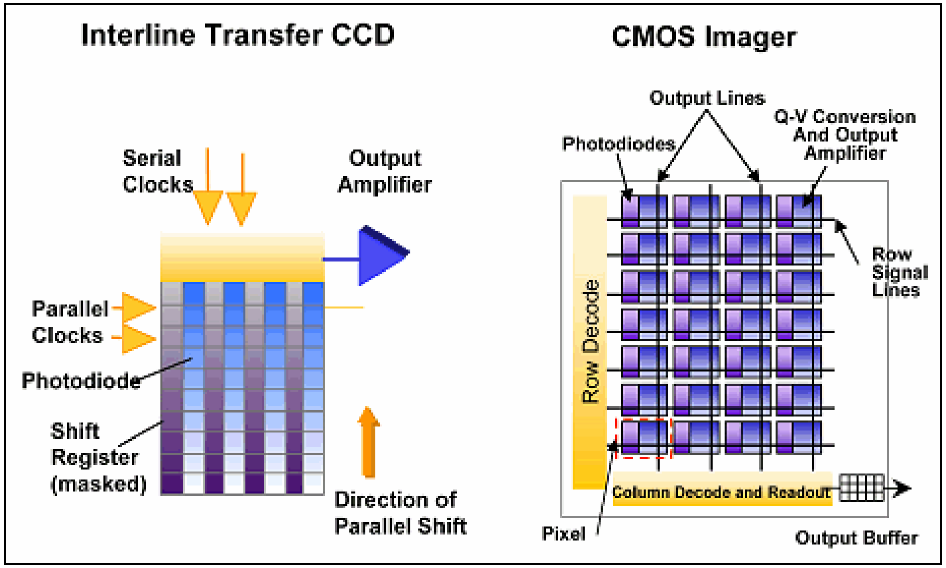


TABLE 1

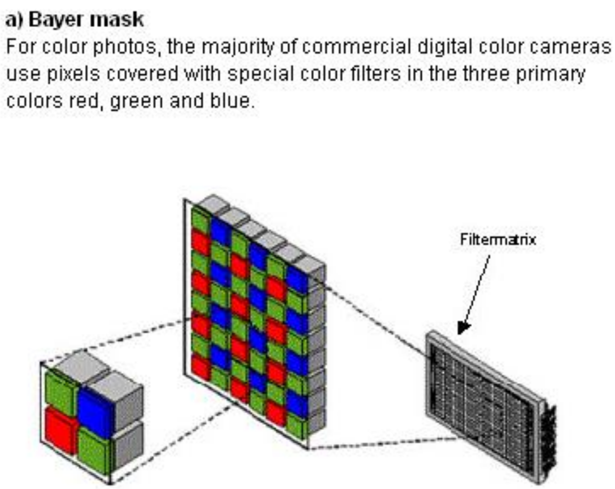
Comparison of CCD and CMOS Image Sensor Features

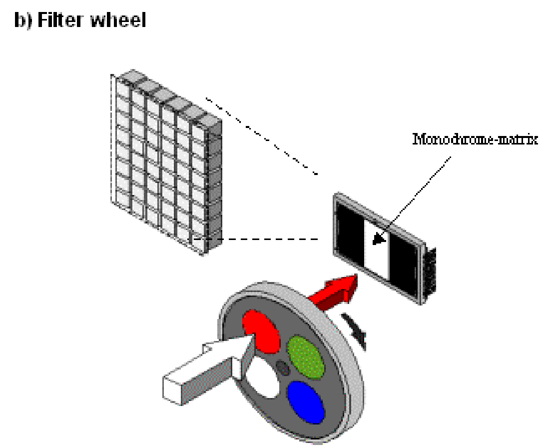
CCD	CMOS
Smallest pixel size	Single power supply
Lowest noise	Single master clock
Lowest dark current	**Low power consumption**
~100% fill factor for full-frame CCD	**X, Y addressing and subsampling**
Established technology market base	Smallest system size
Highest sensitivity	Easy integration of circuitry
Electronic shutter without artifacts	

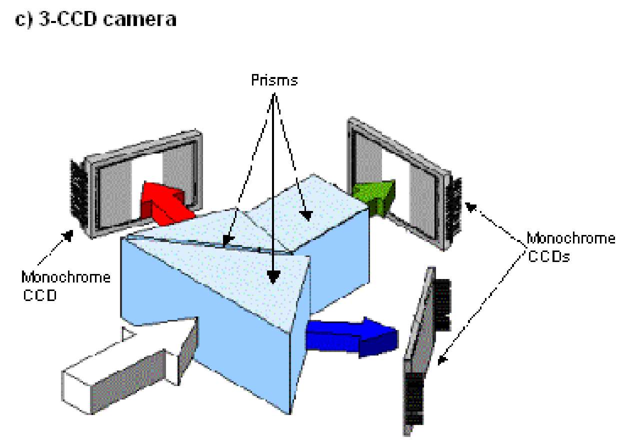
Imaging sensors

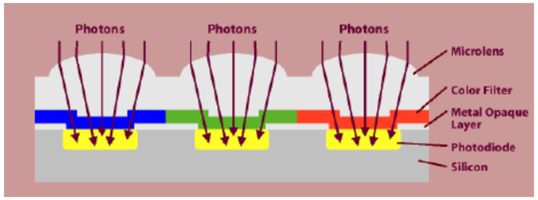
Color imagers

http://www.siliconimaging.com/RGB%20Bayer.htm
http://www.zeiss.de/c1256b5e0047ff3f/Contents-Frame/c89621c93e2600cac125706800463c66



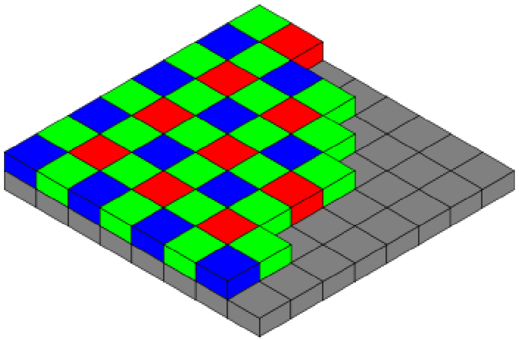


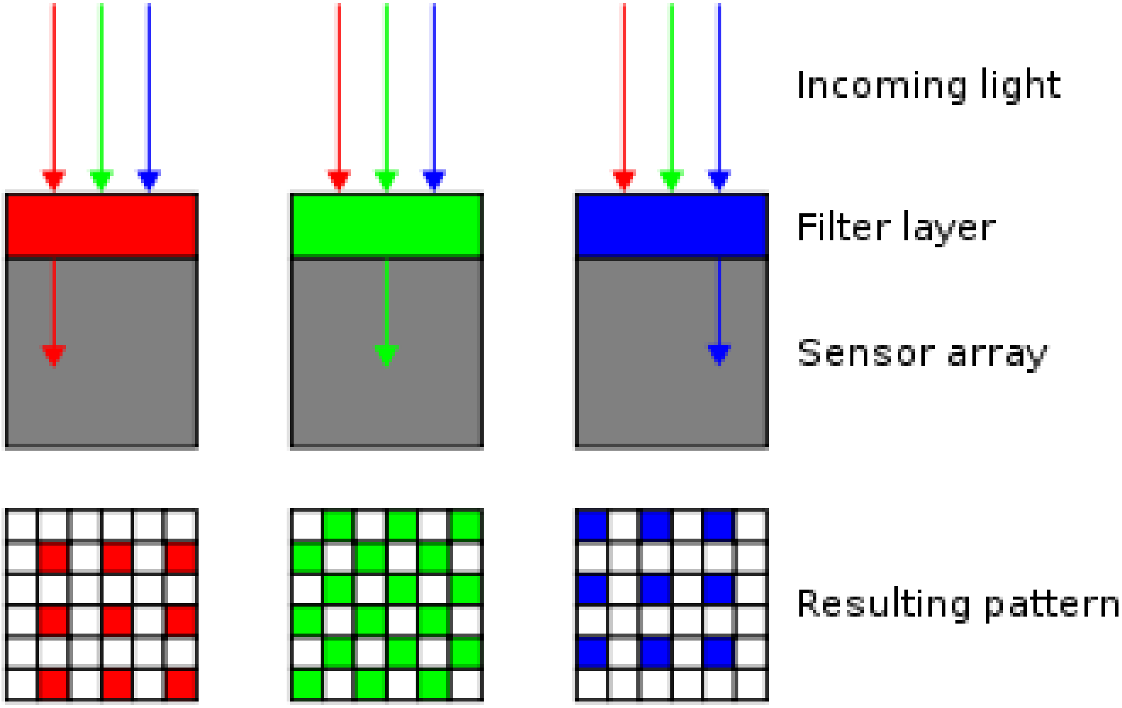




Imaging sensors

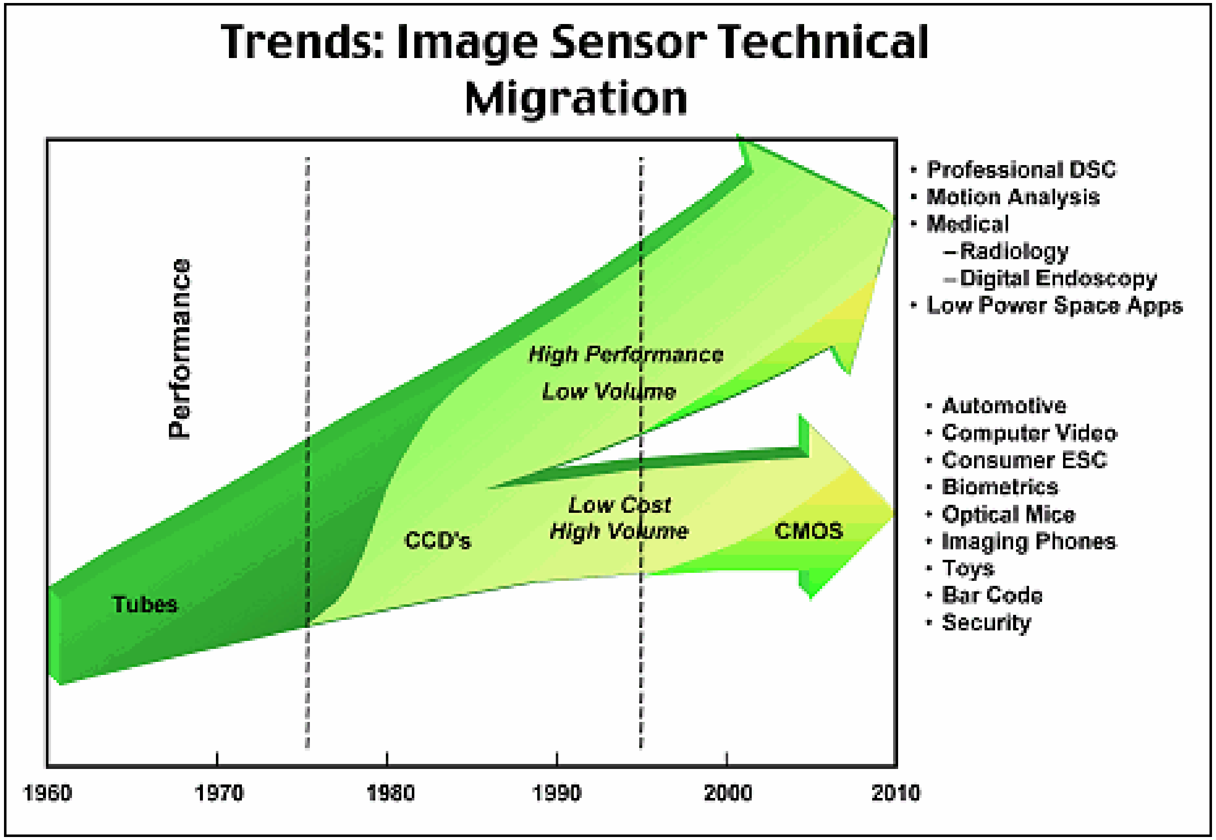
Demosaicing Bayer pattern:
Bilinear interpolation





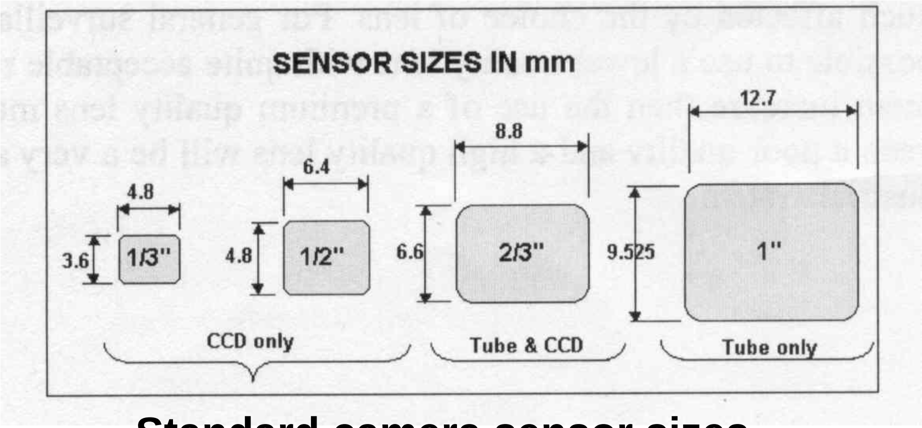
$$G = (G_n + G_w + G_e + G_s)/4$$
$$R_4 = (R_{nw} + R_{ne} + R_{se} + R_{sw})/4$$
$$R_{2c} = (R_n + R_s)/2$$
$$R_{2l} = (R_w + R_e)/2$$

Imaging Sensors

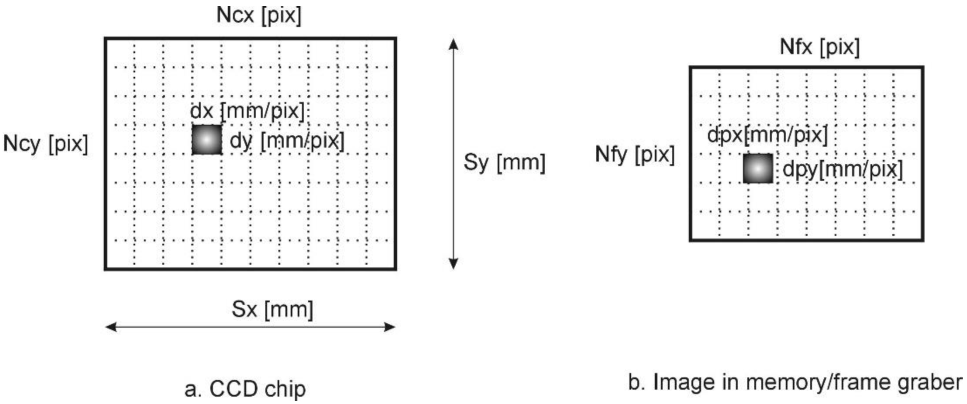


Imager parameters

Imager (sensor) parameters



Standard camera sensor sizes



Parameters of the imager and image in memory

Imager /image parameters

Sensor parameters:

Sx – width of the sensor chip [mm]
Sy – height of the sensor chip [mm]
Ncx – number of sensor elements in camera's x direction;
Ncy – number of sensor elements in camera's y direction;
dx – center to center distance between adjacent sensor elements in X (scan line) direction;

$$dx = Sx/Ncx;$$

dy - center to center distance between adjacent CCD sensor in the Y direction;

$$dy = Sy/Ncy;$$

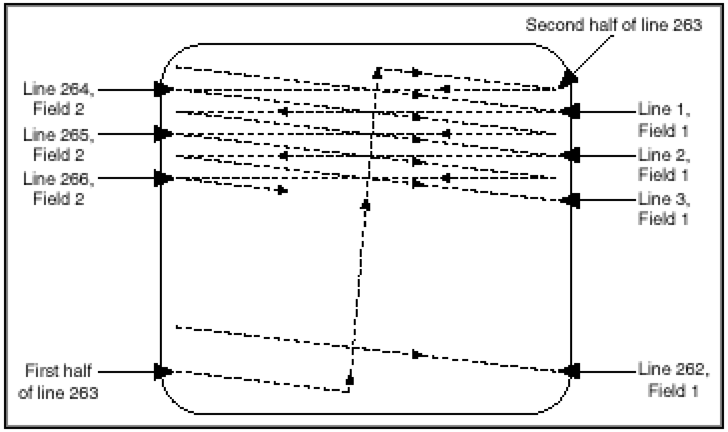
Image parameters (related to the image in memory/framegrabber):

Nfx – number of pixels in x direction as sampled by the computer;
Nfy – number of pixels in frame grabber's y direction
dpx – effective X dimension of pixel in frame grabber, $dpx = dx^* Ncx / Nfx$;
dpy – effective Y dimension of pixel in frame grabber, $dpy = dy^* Ncy / Nfy$;
Ncx / Nfx – uncertainty factor for scaling horizontal scanlines;

Image scanning

Scanning Techniques

- **Determined by application area**
- ***TV imposed the interlacing* of two successive images read from a standard camera**



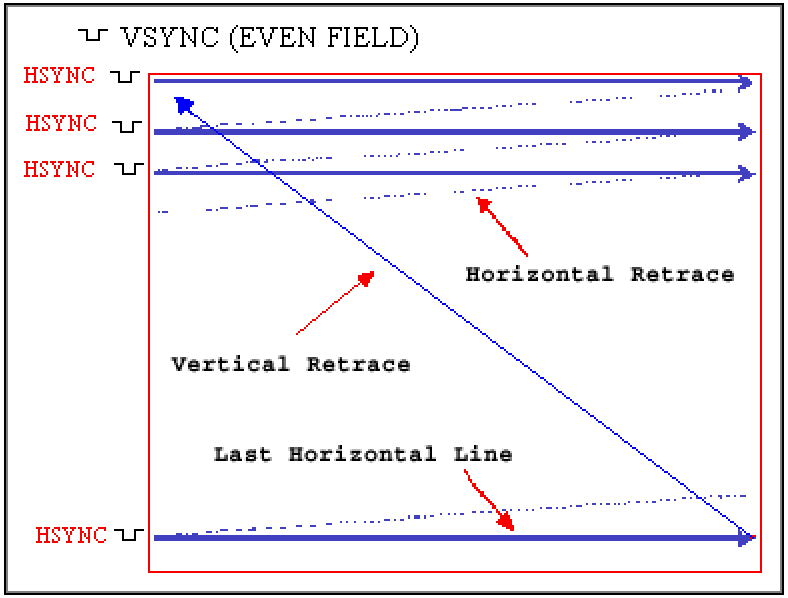
2:1 Interlaced Scanning, shown here for the NTSC Video Format. PAL and SECAM Interlacing is similar, with the difference being the number of lines in each field

- Read/display all even-numbered lines (even field, half-size)
- Restart
- Read/display all odd-numbered lines (odd field, half-size)
- Stitch the even and odd fields together and form a single, full-size frame
- Output the full-size frame

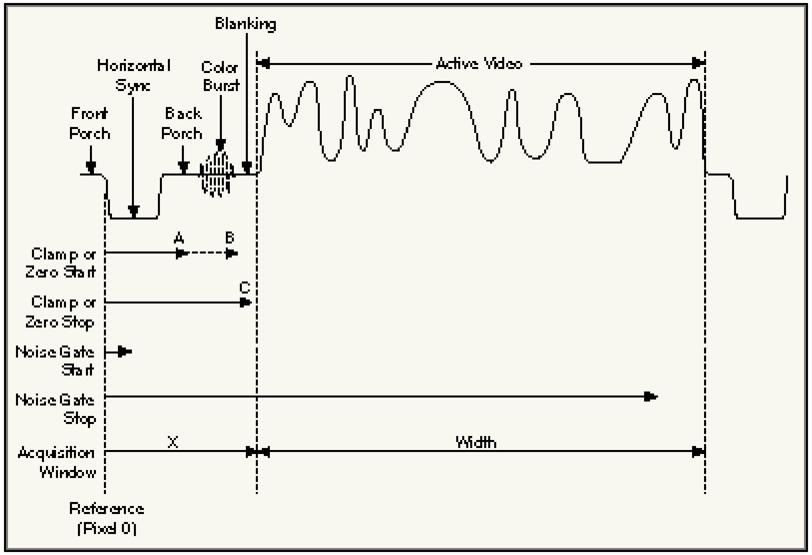
- even and odd frames, full frames
- NTSC Video format: 30 frames/s (525 lines/frame)
- PAL Video format: 25 frames/s (625 lines/frame)

Video signals

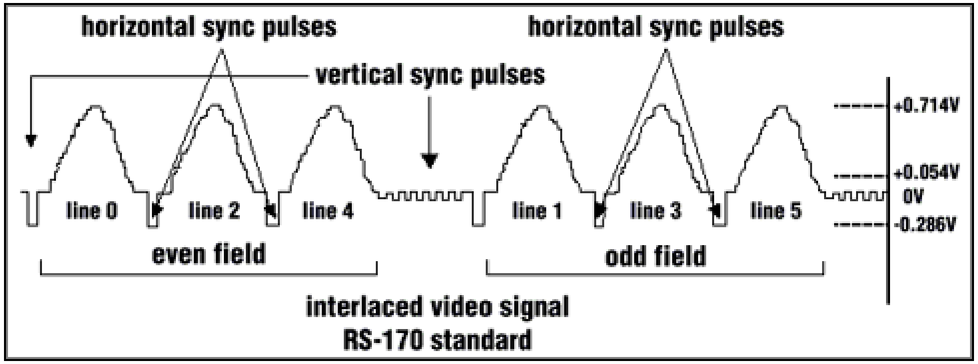
- Analog video signals



The incoming video signal conversion in individual pixel values for displaying



Analog video signal components



Standard monochrome signal

Analog standards

Format	Country	Mode	Signal Name	Frame Rate (frame/sec)	Vertical Line Resolution	Line Rate (lines/sec)	Image Size (WxH) pixels
NTSC	US, Japan	Mono	RS-170	30	525	15,750	640x480
		Color	NTSC Color	29.97	525	15,734	
PAL	Europe (except France)	Mono	CCIR	25	405	10,125	768x576
		Color	PAL Color	25	625	15,625	
SECAM	France, Eastern Europe	Mono		25	819	20,475	N/A
		Color		25	625	15,625	

Parameters of interest:
- # of lines/frame: 525 (this includes 485 lines for display; the rest are VSYNC lines for each of the two fields)
- line frequency: 15.734 kHz
- line duration: 63.556 microsec.
- active horizontal duration: 52.66 microsec.
- # active pixels/line: 640

Digital video signals

Diagram 9: Equivalence between analog composite and digital video

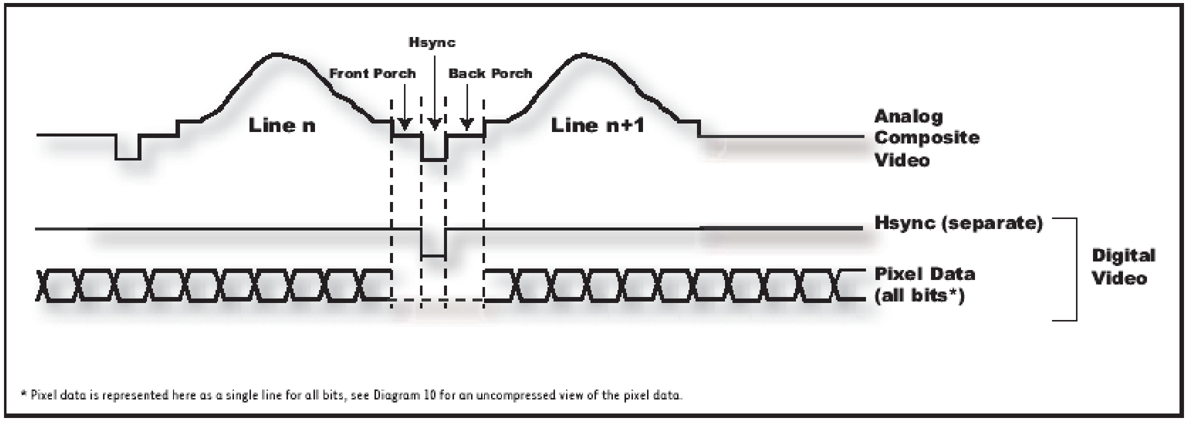
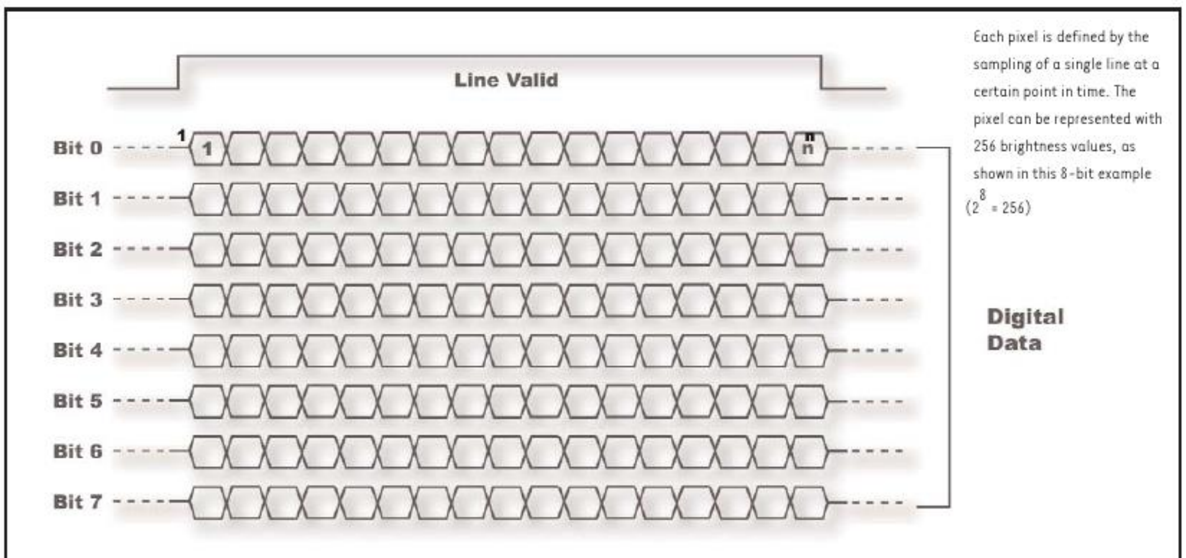


Diagram 10: 8-bit digital video.



Digital transmission standards:

- **Camera Link: 1.2Gbps (base) ... 3.6Gbps (full)**
- **RS 422 / EIA-644 (LVDS): 655Mbps**
- **USB 2.0: 480 Mbps**
- **IEEE 1394: 400 Mbps**
- **USB 1.1: 12 Mbps**

Technical University of Cluj Napoca
Computer Science Department