vision robotics. | Binocular stereopsis [illegible]

and Depth Perception. | Naive, epipolar matching | EM(N^3) done by 2 view geometry

imaging constraint | Stereo depth : each 2D pt gives a 3D line. ∴ No Depth by 1 img. (7)

Point in I1 epipolar line in I2. $O(N^3)$. ← E matrix. If P1 not in

I2 discard ∵ Threshold. | Epipolar matching algo :- 1) Calibrate

cameras i) K1, K2, R_1, R_2, t_1, t_2 ii) Get R t :- $R = R_2 R_1^T$, $t = t_2 - R_2 R_1^T t_1$

2) Get E. $E = [t]_\times R$ | Epipolar constrain $\hat{x}_2^T E \hat{x}_1 = 0$ | $\hat{l}_2 = E \hat{x}_1$

$\hat{x}_2^T \hat{l}_2 = 0$ ← second pt lies on second epipolar line. $\hat{x}^T \hat{l}_2 = 0$. || NORMALISED

$\hat{l}_1 = E^T \hat{x}_2$ | $\hat{l}_1^T \hat{x} = 0$: P1 on line 1. || Epipolar Rectification

2 virtual cameras having diff. K_v & R_v. K : avg of K_1 & K_2. R = interpolate half way. $t_1 != t_2$. $t_{virtual} = R_{vir} R_{real}^{-1} t_{real}$.

H (homography) $= K_v R_v R_R^T K_R^{-1}$ || warping :- $x_v = H x_R$; eg :- Scale 2.

1) Forward 2) Backward, $x_R = H^{-1} x_v$

Interpolation method : Nearest neighbour, Linear, Cubic. || Polar rectification : used when epipole inside camera (Robot Front).

Establishing Correspondences : 1) Feature based (sparse) (corners) 2) Pixel based (dense) (Entire region) || Pixel based (Too much ambiguity) hence we do window match. || min max disparity, search window.

Match metrics :- 1) SSD : $\sum_{u,v} (I_1(u,v) - I_2(u+d, v))^2$ Sum of Square Dist.

(Equal brightness) 2) NCC : Normalized Cross Correlation.

$$\frac{\sum_{x,y} (w_1(x,y) - \bar{w}_1)(w_2(x,y) - \bar{w}_2)}{\sigma_{w_1} \sigma_{w_2}}$$ $\bar{w}_i$: mean. $\sigma_{w_i} = \sqrt{\frac{1}{n} \sum_{x,y} (w_i - \bar{w}_i)^2}$

÷ by variance ∴ robust to change in Lighting & camera Response func.

3) NSSD (Normalized SSD) : Same to NCC. (8) ☆

$$\sum_{u,v} \left(\frac{I_1(u,v) - \bar{I}_1}{\sqrt{\sum_{u,v} (I_1(u,v) - \bar{I}_1)^2}} - \frac{I_2(u+d,v) - \bar{I}_2}{\sqrt{\sum_{u,v} (I_2(u+d,v) - \bar{I}_2)^2}} \right)^2$$

Note : Infrared to visible matching : Info theory maximize mutual info.

Disparity ≈ Depth map. | Drawbacks : 1) Multiple interpretation (Epipolar ordering) : [illegible] 2) Ambiguity : Repeating patterns 3) Window size

i) Small : low ambiguity, high noise. ii) Large : fine detail lost, high disparity [illegible]. 4) Lighting cond. 5) Half occluded.

ii) Window shape : ground level robotics.

|| Stereo constraint : 1) 1D epipolar search 2) Monotonic ordering (L2R) 3) Image brightness constancy 4) Match uniqueness 5) Disparity continuity 6) Disparity limit (search range) 7) Fronto || surfaces (slope) 8) Feature similarity 9) Struc. grouping. || 3D Reconstruction w/o epipolar rect : line won't intersect ∵ noise || 3D with epipole :- C_1, C_2, X is on one epipolar plane.

Z(depth) = (baseline(d) × focus(f)) / disparity ($X_L - X_R$) || Challenges : 1) Photometric 2) Surface (lack / repeating) 3) Geometric (occlusion) || Fixes :- 1) Trinocular (3 camera, ext to P to P) 2) Helmholtz Reciprocity Stereopsis $i_R = \rho(\hat{v}_r, \hat{v}_l) \cdot (\hat{n} \cdot \hat{v}_r) / |p_r - p|^2$; SWAP(L,R)

[High latency, reversible BRDF, pt source illumination

Motion constraint : works on velvet too.

Uncalib 3D recons. (upto a 3D Proj. scale) $\hat{x} : K^{-1} x$ [$\hat{x}$ in norm coord] || $F = K_2^{-T} E K^{-1}$ || F fully [illegible] all E formulas in pixel coord. || Feature extraction : 1) Detection 2) Description : (9)

i) Invariant (# of pixels) ii) Covariant (Area) (Scale 2) || Corner : covariant to pos.

orientation, Invariant to scale : sigma, window fixed || Gaussian Pyramid : Blur → Sub sample. Binomial kernel (Rate 2) || Scale Space Blobs : $2^{(level-1)}$, objects < 2σ are removed | Level of [illegible]

[illegible] changes. SD = ½ width, image DxE. || Fast move obj : G Pyr. || SIFT (Scale invariant feat. transform)

features = keypoints ; Invariant : Scale, Rot | Robust : viewpoint, illumination, Noise || [illegible]

SFM : 2+ img stereo. 2 Approaches : 1) Wide baseline (discrete) 2) Infinitesimal (conts.) (sub pixel movement) || After KNN, object recog. → min. spanning tree. || 2 view / Pairwise est. : 1) Greedy Algorithm (Give local best) (Small motion) 2) Otherwise Sift + Ransac (Big Diff).

2 view calib. algo (we have K) : 1) Feat detec. 2) Determine feat. corres. 3) Compute E mat 4) Get relative R & t, only up to scale (only dir.) unlike full calib stereo. (10)

5) Sparse / dense reconst. || 2 view uncalib (No K too) : Can compute F & only upto 3D Proj. : F : 7 dof (Proj recons : X(9 - 1 rank 2, singular det = 0, -1)

E : 5 dof (R+t -1), Similarity recons.

E : 5 point corres. → 10 real ; 6 Ps → 1 real ; 7 Ps → 1 or 3 real, 2 or 0 complex ; 8 Ps → 1 real.

F : 7 Ps same ; 8 Ps → 1 real || E : Linear estimation $Ae = 0$, solve for e, Right null space of A. Do SVD, rightmost of V is your E. Now, to do rank 2, Do SVD, diag = (1, 1, 0). Then get final E. U : m×m, Σ : m×n, V : n×n. To get R, t pick front one from 4 solutions. || F full same, end : diag = (σ_1, σ_2, 0) || Uncalib camera projection $P_1 = [I | 0]$, $P_2 = [[e']_\times F | e']$. Retrieve epipole from F :- $F e_1 = 0$ | $e_2^T F = 0$ | $F^T e_2 = 0$ || Normalizing F : 1) Determine centroid 2) Get standard deviation 3) Scale pts to SD = 1 4) Translate to origin.

3 views : tighter constraint (P-P-P) (Trifocal plane). Trifocal tensor (3 × 3 × 3). 3 Proj : 36 − scale 3 − 3D Proj (15) = 18 DOF. Tensor contraction = Matrix vector Multiplication. || Einstein summation : Reduces vec. by 1D.

M-view [illegible]

assuming that all 3D pts are vis. in all imgs. For Calib :- 1) 3D struc. : 3n 2) Cameras 1st Mono Cam = P = [I | 0] ; Rem rot. : 3(m−1)

Rem trans 3(m−1) − 1 || Total unknown : 3n + 6(m−1) − 1.

Soln exist if unknown <= known = 2nm. For Uncalib 2) Camera unknowns = 11m − 15 ∴ Soln 3n + 11m − 15 <= 2nm || N view reconst. Use linear est. from 3 point, Marquardt (diag) here and converge to minima g(R, t). (Iterative algo) Non spanning tree.

MODEL fitting : Reject outliers || Hough Transform : x, y → m, b. Hough space. pt → line, line → pt. || dist $d = x \cos\theta + y \sin\theta$

Algo : For every (x, y) pt, for θ 0 → 180, find d. ⇒ H[d, θ]++. (Voting algorithm). Max votes bin represents the fit line.

Cons : Hogs to noise. || Generalize Hough transform. Add n no. of params to repres any curve. Circle : $(x-x_c)^2 + (y-y_c)^2 = r^2$

Algo : Given all the pt x, y, compute r, H[[illegible]]++. Eg. KLH, we detect circle, Remove, detect □, remove, done.

Orthogonal Regression (Line fitting) : Minimize : E(a, b, d) $= \sum_{i=1} (a x_i + b y_i - d)^2$ wrt a, b, d. (On PD) $d = \frac{1}{n} \sum_{i=1} a x_i + b y_i = a\bar{x} + b\bar{y}$. ∴ $E = \|U n\|^2$, $n = (a\ b)^T$ (11)

$$U = \begin{bmatrix} x_1 - \bar{x} & y_1 - \bar{y} \\ \vdots & \vdots \\ x_n - \bar{x} & y_n - \bar{y} \end{bmatrix}$$ $E = \|U n\|^2 = n^T U^T U n = n^T S n$. $$S = \begin{bmatrix} \sum x^2 - n\bar{x}^2 & \sum xy - n\bar{x}\bar{y} \\ = & \sum y^2 - n\bar{y}^2 \end{bmatrix}$$

Real, Sym, +ve def. → By Lagrange : $S n = \lambda n$. n : right most col of V (smallest diag) || Ransac : Can be used for any parameterizable model. || v λ v Rel "betw" SVD, EVD Do $A^T A$, $U^T U$ gets cancelled λ = sigma². || Ransac : Random, Fit (), Consensus set, Max inliers model wins.

Dist for consensus set : 1) L^2 dist to epipolar line 2) Epipolar constraint 3) Reprojection error or sampson err approx.

∵ Err ortho dist be good in one dir, we do $E_{tot} = d1 + d2$. OR $d = q_2^T F q$ || $N = \log(1-p) / \log(1-(1-e)^s)$ || Thresholding $T = (1-e)$: fraction that [illegible]

N : tot. no. of samples || Prob. of inlier α, p : prob. 1 sample is free from outliers. If α = 0.95, σ : Gaussian || Proj Mat : 5.99

For E, F, 2D line dist thresh $t^2 = 3.84\sigma^2$

Optical flow Motion : Quiver plot = motion field. What causes motion : 1) obj moves 2) Camera moves 3) obj articulate 4) Bend / deform 5) Smoke / clouds 6) Combi of up (Multiple) || With FOE we can calc 1) dirn of motion 2) Time to collision. FOE : Intersec. of vel. vectors. $\dot{X} = -(\omega \times X + t)$ (12)

$\dot{x}$: how obj moved. Motion field pixel coord :-

$$\dot{x} = \frac{t_3 x - t_1 f}{Z} - \omega_2 f + \omega_3 y + \frac{\omega_1 xy - \omega_2 x^2}{f}$$ $x = \frac{f}{Z} X$

$$\dot{y} = \frac{t_3 y - t_2 f}{Z} + \omega_1 f - \omega_3 x + \frac{\omega_1 y^2 - \omega_2 xy}{f}$$ $y = \frac{f}{Z} Y$

Pure translation ω = 0, 1) Radial [illegible] 2) Parallel (FOE at ∞) FOE = 0

Pure Rot : t = 0 || Depth(Z) ∝ 1 / img velocity. || Measure Motion : (Optical flow) 1) Feature based (Detect, track) 2) Differential tech.

Motion field : Projection of velocity vectors || Assumption : 1) Const. Color 2) Small motion. || $\frac{dx}{dt}\frac{\partial I}{\partial x} + \frac{dy}{dt}\frac{\partial I}{\partial y} + \frac{\partial I}{\partial t} = 0$ || ∂I/∂x using G x, using G y & ∂I/∂t using [−1, 0, 1]. || Increase aperture to get more info and solve for dx/dt, dy/dt || 2 generic ways to get flow : 1) Think globally & regularize over image, 2) Look over window & assume const. motion || Horn & Schunck algo : Smoothness constraint : $e_s = \iint ((u_x^2 + u_y^2) + (v_x^2 + v_y^2)) dx dy$

OF constrain eq term : $e_c = \iint (I_x u + I_y v + I_t)^2 dx dy$

Minimize $e_s + \lambda e_c$ || Lucas Kanade : Assume const. vel. (u, v) within an image window Ω. $\begin{bmatrix} \sum I_x^2 & \sum I_x I_y \\ \sum I_x I_y & \sum I_y^2 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} \sum I_x I_t \\ \sum I_y I_t \end{bmatrix}$

When M is singular : 1) ∇I = 0 2) Window = 1 Px 3) At edge. ∇I [illegible] M^{-1} valid at corners (Rich texture) || How to fix : 1) Iterative refinement : keep doing u, v until 0 2) Coarse to fine : 2 Pyramid, Repetitive I^k at each level. = Multi resolution LK. Vid compress. obj detec & tracking >>> OF.

Recognition challenges : 1) Within class variability 2) Pose / Perspective (R, t) variability 3) Lighting 4) Occlusions 5) Clutter in BG.

|| Confusion Mx, FN : [illegible], FP : [illegible] (13)

TPR + FNR = 1 ; FPR + TNR = 1

Recall = TP / (TP + FN) ; Precision = TP / (TP + FP)

SVM : optimal hyperplane ; It uses support vectors to get that plane. (Auto avoid outliers) (Greedy) ; separating (OSH)

Bag of features : extract features, put into comparable histogram. VISUAL VOCAB

KNN = Template matching | Distance func^n L1, scalrest, etc. L2: Sqrt sq. is hardware intensive, hence ssd. KNN costly as [illegible]
Curse of dimensionality. Linear Subspace: 3D to 2D. A k dim subspace is spanned by k linearly independent vectors. $y = Wx$
y: Low dim, x: High D. How to choose W: 1) Drop dim. manually 2) Random projections (enforce orthogonal constraint on W) 3) PCA 4) LDA Linear discriminant analysis. (Lisadore) 5) Independ. CA 6) Non-linear dim red.
PCA: label, LDA: label. Covariance $\Sigma_x = \frac{1}{n-1} \sum (x_i - \bar{x})(x_i - \bar{x})^T$
Mean: $\mu_x = 1/n \sum x_i$ || Col. of V = eigen vector = PC coeff. Mean Devi^n form: $\begin{bmatrix} x_1^T - \mu_x^T \\ \vdots \\ x_n^T - \mu_x^T \end{bmatrix}$; $\Sigma_x = \frac{1}{n-1} \tilde{X}^T \tilde{X}$; $\hat{X} = \tilde{X} V$ (Proj^n) ; $\hat{X} V^T = \tilde{X}$ (Back N)
SVD properties: Col space = cols, Null space (m−r, left, U) (n−r, right, V)
r = rank (A) = # non zero singular val
PCA for recog^n :- Biased est: consistent, unbiased: finite obs.
1) Vectorize data 2) Mean deviation Form, Covariance M x 3) svd(cov) OR svd ($\tilde{X}$) 4) Get k components, 5) k comp. × $\tilde{X}$. Challenges :- 1) Small variances lost. 2) Hence, underequipped w/out labels fast will be fooled. LDA: Better w. Algo: 1) Get global mean 2) mean for each class 3) Compute between class scatter. $S_B = \sum |\chi_i| (\mu_i - \mu)(\mu_i - \mu)^T$ 4) Within class: $S_W = \sum_{i=1} \sum_{x_k \in X_i} (x_k - \mu_i)(x_k - \mu_i)^T$ Total Scatter: $S_B + S_W$
Generative can [illegible] model, // Projec^n, Affine. (14)

Neural Net: Main aim: To create hyper plane for non linear sig. (15)
$y(x;w) = \sigma(w^T x + w_0)$ | Remember bias term | 1) tanH: active func^n (−1 to 1) 2) Rectified linear unit (ReLU) :- Zero till 0, = to the O/P value otherwise (>0). Easier ∴ Back Prop. use derivative & dg/dz = 0 or 1 only here. || To classify: Soft max layer at end to represent values in PDF. Each & every problem has a linearly separable space by just ↑ing the no. of layers. (UAT) (Universal approx theorem). Challenges of UAT: 1) won't find params to train 2) May Overfit 3) ANS become too large to compute. || Training is minimizing loss func^n. Loss func's: 1) Regress^n: L1, L2 norm 2) Classific^n: softmax + cross entropy (dist betw^n PDF) || Random init to start. || How to tune weights: 1) BP: stochastic grad. descent 2) Adagrad, Adam, 3) Regularization 4) Early stopping. Back Prop :- We propagate the err back, do grad. descent and apply wts. || End P called Inference || Avoid overfit by 1) Use valida^n data, 2) Regularizat (Makes it generalized model) 3) Dropout: only trains selective w at a time, 4) Early Stopping. 5) Data augment^n : Use jittering (Synthetic dataset)
Deterministic: Use full train data | Stochastic: Use subset of train data (Mini Batch size. SGD algo: 1) Choose a learning rate 2) Stop criteria 3) mini batch size 4) Random select a mini batch 5) Use Eval & [illegible], 6) Update eq^n - (Repeat)
SGD with momentum :- 1) Get gradient 2) Compute velocity 3) Update $\theta_{t+1} = \theta_t + v_t$ ∝ Tex η decreases.

Problem with FC: 1) Space heavy 2) Vanishing gradient. (16)
To reduce complexity: 1) Spatially local regions. 2) Convolution kernel as weights. Hence "Shared wts." Conv (inner prod.) || For full conv. you padd the image. || Pooling: Math op^n on prev. o/ps. Max, Min, etc. This reduces no. of nodes. Also has stride || All channels filter indepen. 3 conv kernel = 9 o/p in (RGB) × 3. Output = (Input dim + I/P Pads T/B or L/R − Kernel Ht or width) / stride Ht or width + 1 | FC layers at end to make feat. vector. | CNN > FC ∵ spatial struct^n helps in localiz^n of objects.
Domain Transfer: Down pretrain, Cut Paste, Lock train, Unlock train DONE.

Params $\sum$(Kernel × kernel × 3D I/P × Depth + Depth) |||| Photometric Stereo: 1) Shape from shading (Known light & BRDF, Restrict) 2) Photometric stereo: Mul cam, mul light (Lamb. assume) (4)
S: 1) General BRDF 2) Lamb. BRDF 1) unknown light | BRDF: Bi-dir reflectance dist func^n, $P(\theta_i;\phi_i;\theta_o,\phi_o)$
Ratio of radiance to irradiance || Gradient space (p, q) $p = \partial f/\partial x$; $q = \partial f/\partial y$; $n = \partial s/\partial x \times \partial s/\partial y$
$= (-p, -q, 1)^T$ $\hat{n} = n/\sqrt{p^2+q^2+1}$ || Irr. func of only n: $= E(p,q) = Lp^{1+}...$ [illegible] | Intensity / E of one pixel constraints the surface normal projecting that pt to a curve. (3 to triangulate the normal dir) || Recover surface from n.
$p = -n_x/n_z$, $q = -n_y/n_z$ | 1) Integrate p along row 0, q along all cols using S 2) Use convex mask 1) Normalize (sub. min.) ∵ We don't have the constant of integra^n. || Cons not perfect: data is noisy (O) || Lamb. known light: $I_i(x,y) = a(x,y)\,\hat{n}(x,y)^T s_i$, $s_{i,0}$ $\hat{s}_i$ || a intensity, irradiance
$b(x,y) = a(x,y)\,\hat{n}(x,y)$ | $a(x,y) = \|b(x,y)\|$ | $\hat{n}(x,y) = b(x,y)/\|b(x,y)\|$ | $b(x,y) = S^+ I(x,y)$
$S^+ = (S^T S)^{-1} S^T$ || n ≥ 3 light srcs || $I_{SDV} = [R] I_{RGB}$ || $R = \begin{bmatrix} \hat{c}^T \\ [\hat{c}]^\perp \end{bmatrix}$ $[\hat{c}]^\perp$: Left null of c, $= \sqrt{u^2+v^2}$
if det(R) = −1, negate the last row. | S: specular | D: diffused

Image filters: o/p is weighted sum of nearby pixels | Linear filter: output is linear func^n and is shift invariant of i/p. Correla^n $w(s,t) ☆ f(x,y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s,t) f(x+s, y+t)$ | 180° rotate if convo (−) (Symmetric C=C) || Padding Size 0/1 (5)
Shape :- 1) Full $P = A + 2(C-1)$, $Q = B + 2(D-1)$, $O_B = B+D-1$, $O_A = A+C-1$
2) Same: $P = A+C-1$, $Q = B+D-1$, $O_B = B$, $O_A = A$ 3) Valid $P = A$, $Q = B$
$O_B = B-D+1$, $O_A = A-C+1$ | Unbias filter: Sum of wts = 1. | This is done to maintain the 0-255 range | Sharp filter: unbiased: scale up then subtract mean.
If we dot kernel with similar Mx, it amplifies o/p hence Template matching.

Prop of Conv: 1) Commutative $a * b = b * a$ 2) Associative $a*(b*c) = (a*b)*c$
3) Distributive $a*(b+c) = a*b + a*c$ | Correla^n only dist.
4) Scalar 5) Identity. Prop. of cont. Conv :- Same + differentiation. Rule: $\partial/\partial x\,(a * b) = \frac{\partial a}{\partial x} * b = a * \frac{\partial b}{\partial x}$ || Types of Noise :-
1) Simple: light fluct., Sensor, Quantiz^n, Finite precision 2) Complex Noise: shadow, blur edges. (Avg) filter same. (Box filter) (called Gaussian) Size of Gaussian kernel = 6σ or 6σ+1 if even. || Frequency domain useful for orienta^n detec^n. More data stored in phase, very less in magnitude. Phase stored in 7.2 ℓi. | Convolution using freq: 1) FFT of both imgs 2) Ele by ele mul 3) Inverse FT | Note :- High freq: small details. | Gaussian → Gaussian | Square → Sync. || Median filter of salt & pepper noise (Nonmetal chop.) [Non linear filter] (Con. Shift) :-
Segmenta^n : 1) Semantic 2) Instance. || Factors for an edge: 1) Depth disc. 2) Surface orienta^n 3) Illumina^n disc 4) Specular 5) Reflectance disc.
Detect edge: smoothing, take derivative. FD: edge at peak, SD: edge at 0 intersec middle. $f'(x) = [f(x+h) - f(x-h)]/2h$; $f''(x) = [f(x+h) - 2f(x) + f(x-h)]/h^2$ || h = 1 for pxl $G_x = [-1/2\ 0\ 1/2]$ $G_y = [-1/2\ 0\ 1/2]^T$ || Edge detec^n threshold → mag: $\sqrt{G_x^2 + G_y^2}$ Dir^n: $\tan^{-1}(G_y/G_x)$ || Combine smooth & gradient by Fullcorr. $\cos\theta\, G_x + \sin\theta\, G_y$ [illegible] | NMS: [illegible] find center, follow into normal. [illegible]
pole: 45° (any edge [illegible]) || Corners: rapid change in curve of img, viewpoint invariant. High gradient ∴ No relevant for single points. Search neighbours along gradient on the curve do NMS || Shi Tomasi: 1) Take image and calculate spatial gradient $M = \begin{bmatrix} \sum G_x^2(s,t) & \sum G_x G_y(s,t) \\ " & \sum G_y^2(s,t) \end{bmatrix}$ (6)
2) Minor eigen val $\lambda_{min} = \frac{Tr(M) - \sqrt{Tr(M)^2 - 4 det}}{2}$
3) Do window NMS get measurement image R. 4) Thresholding or top N.
NMS: helps remove, [illegible] repeating corners.

Color: Affected by: 1) Spectrum of light striking retina 2) Neighbour space 3) Neighbour time 4) State of mind. || Light: RGB, Pigment: Cyan, magenta, yellow. (Subtr.) || Color defined by 1) spectrum 400-700 nm. 2) Names 3) RGB || Color reflectance (BRDF): basically which light is stimulated for that object. || Color signal = illumination * reflectance | Camera response func^n affects. CR is usually a scalar or linear factor. || Spectrometer used for refl. measurement) | Viewing ∠ also effects hence BRDF. Response of Kth Cone $\int P_k(\lambda) E(\lambda)\, d\lambda$ || 3 types of cones: 1) Short λ (Blue) 2) Mid λ (green) 3) Long λ (Red) || XYZ color transition pnt || CIE L*a*b* : Non linear, designed to approx human vision (diff in color perceived) || CIE XYZ / xy (17)
Y: grayscale (illuminance), Z: Blue, X: +ve || $x = \frac{X}{X+Y+Z}$, $y = \frac{Y}{X+Y+Z}$
$z = 1 - x - y$ || $X = (Y/y) \times x$ | $Z = (Y/y)(1 - x - y)$. 4 chromaticities to represent in color gamut: R, G, B, white point (sRGB)
Chromatic Adapt^n: One color to another. | LAB used for day to night light
Luminance: calc from linear RGB (Y), Luma: from gamma RGB

Study Human Vision :- 1) Physiological 2) Psychological 3) Collect [illegible] 4) Func. MRI || Spatial Freq. is cycles / deg. || Lens: Thick - close, Thin - Far | Rods: grayscale || Retinal neuron: 1) Bipolar cell 2) Ganglion. | Dendrite I/P, Axon O/P. Pathways: 1) Parvocellular - Slow, color 2) Magnocellular - Fast
Cues: 1) Shading cue / contrast 2) Subjective contours 3) Global vs local info (Spiral) (1[illegible])

Vanishing point / Horizon (// Rail track) | Why Homo: 1) We can represent pnts at infinity 2) Avoids div. only *. def. = Rank of Mx || Treat $x = \lambda x$. Just convert Homo to In, λ gets cut || $l^T x = 0$; Rem default is column. $Ax + by + c w = 0$. $n = [a, b]^T$ (col) | $d = c/\text{norm}(n)$ | (0,0) Line at ∞ || Hessian form normalize / Unit line eq^n | $x^T l = 0$ x can be Num. of pnts | Eqn [A], A = ... l_1, l_2 = pnt [x_1], x_2 = line | When VP? // lines L^r intersect image plane [illegible] line
$x = PX$ (Pnt Imaging eq^n) (3D→2D) | $x, y, z = f\left(\frac{X}{Z}, f\frac{Y}{Z}\right)$ | $P = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$
For Panorama (only Rot^n) $x' = Hx$... | 2D→2D Homog
Projec^n Mx. (3×3) || R is SO3 special ortho Mx :- Range [−1, 1], row col is unit vector, row ⊥ col, det(R) = 1, $R^{-1} = R^T$ || Yaw Pitch Roll : x, y, z :-
R_z [c −s 0, s c 0, 0 0 1] R_x [1 0 0, 0 c −s, 0 s c] R_y [c 0 s, 0 1 0, −s 0 c]
$x' = Rx + t$ | $K = [\alpha_x\ 0\ x_0,\ 0\ \alpha_y\ y_0,\ 0\ 0\ 1]$ $\alpha_x = f m_x, f m_y, m_x p_x, m_y p_y$
$x = K[I | 0] \begin{bmatrix} R & t \\ 0^T & 1 \end{bmatrix} X \Rightarrow x = PX \Rightarrow x = K[R | t]X$ | [illegible]

$1/z' - 1/z = 1/f$ || Noises in sensor: Thermal, electronic || Radiance: Power of light travelling in a specific dir. $L(x,\theta,\phi) = P/(dA\cos\theta)\,d\omega$
Irradiance: All light power arriving at a surface / pnt $E(x) = \int L(\ldots)\cos\theta\ldots$
$I = R[\int\int\int\int E(x,y,\lambda,t)\, s(x,y)\, q(\lambda)]$ || $I = k_r k_s L$. (Ambiguity / I ∝ R)
Color: 1) Filter wheel 2) Prism method 3) Filter Mosaic 4) CMOS || BRDF ratio of emitted radiance to incident irradiance. || Specular Reflec^n: $\omega_o = 2(\omega_i \cdot N)N - \omega_i$ incident ∠ [illegible]
Lambertian surface $\cos^+\theta = \max(0, \hat{n}^T\hat{s})$ | Lambert's cosine law
Ambient light: Global illumina^n, Nothing dark. || Phong reflectance model: lobe on right || Isotropic one way velvet, An: 2 way. || Gonio 3D, Gonio 4D
$e(x,y) = s_{a,0}\, k_a(x,y) + s_0 (k_d(x,y) f_d(x,y) + k_s(x,y) f_s(x,y))$ || $f_d = \max(0, \hat{n}^T \hat{s})$
$f_s(x,y) = \max(0, \hat{n}(x,y)^T \hat{h}(x,y))^{\alpha(x,y)}$ || α: shininess of surface facet.
$\hat{h} = \frac{1}{\|h(x,y)\|} h(x,y)$ || $h(x,y) = \hat{s}(x,y) + \hat{v}(x,y)$ (Half L)