1. Viola Jones Face detection-Feature Extraction(harr like)&Classification,*no need to consider conv with inv* Feat.Ext.-1.Kernels(-+edge,-+-line, $\frac{-+}{+-}$4rect feat)2.Convolution with scaled masks3.Descriptor-24x24-160K size *Training* - ① 5K +&-ve(hard -ve, nonface)24x24imgs② Feat. Ext. - 10K des. of 160K len. ③ Classification . *Adaboost* *Adaboost* - ①Strong classifier=combine a lot of weak classifiers②160K feat=160K weak classifiers h_i③Wrongly classified = more weight $[w_{ti} = 1/2m]$ - m-no. of faces, i - index of the image, t- index of the descriptor-**Training**①$Normalise - W_{t,i} = W_{t,i}/\sum_{j=1}^{N} W_{t,j}$②best weak classifier-$\varepsilon_t = \min_{\{\theta\}} \sum_{i=1}^{N} \omega_{t,i} \left|h\left(x_i, \theta\right) - y_i\right|$ ③ $\omega_{t+1,i} = \omega_{t,i}\beta_t^{(1-\alpha_i)}, \alpha_i = 0$,if correctly classified ④Strong classifier - if $\sum_{t=1}^{T} \gamma_t h_t\left(x_{test}\right) \geq \frac{1}{2}\sum_t^T \gamma_t$ where, $\gamma_t = \log\frac{1}{\beta_t}$. *Testing* - Increase size by 1.25 check,*Integral image* $ii(x,y) = ii(x-1,y) + s(x,y); s(x,y) = s(x,y-1) + i(x,y)$, Cascade of weak classifiers -stage consists of few weak' , reject non-faces soon.-start 2feature strong classifier, minimise false negatives-lower threshold - high detection rates false +ve rates

2. Features- pixel intensity not reliable- feature; light, occlusion, angle = Feature - gradient, HoG, SIFT etc.①*HOG*-Oriented gradient-vectore $m = \sqrt{I_x^2 + I_y^2}, \theta = \tan^{-1}\left(\frac{|I_y|}{|I_x|}\right)$ - angle is oriented w.r.t to the signs of $I_x I_y$ - note y is from top to bottom of image

3. SIFT①*Scale Space Extrema*- no.of octaves = 4, number of scale levels = 5, initial$\sigma = 1.6, k = \sqrt{2}$ scale space= L(x, y, σ), don't confuse with scale in σ with image resizing, DoG is subtracting each guassian scale images, 5 scale levels=4 DoG images, extrema found within the octave for different guassian scales, 2nd octave - resample Gaussian image - 2σ(2 image from top) evry 2nd pxl in r and c. *DoG - difference of gaussian*-$I * (G_{\theta_1} - G_{\theta_2})$ *LoG - Laplacian of gaussian*-$\Delta G = \nabla^2 G = \nabla \cdot \nabla G = \frac{\partial^2 G}{\partial x^2} + \frac{\partial^2 G}{\partial y^2}; LOG(I) = I * \Delta G$ ②*Keypoint Localisation* - fit nearby for location, scale, and ratio of principal curvatures 2.1*ContrastThres* 1. Noise correction - $Taylor - f(x) = f(a) + f'(a)(x-a) + \frac{f''(x-a)}{2!}(x-a)^2 + \ldots; D(\mathbf{x}) = D + \frac{\partial D}{\partial \mathbf{x}}^T \mathbf{x} + \frac{1}{2}\mathbf{x}^\mathbf{T}\frac{\partial^2 D}{\partial \mathbf{x}^2}\mathbf{x}; \left.\frac{\partial D(\bar{x})}{\partial \bar{x}}\right| = \left[\begin{array}{c} \partial \mathbb{D}/\partial x \\ \partial \mathbb{D}/\partial y \end{array}\right]; \frac{\partial^2 D(\bar{x})}{\partial \bar{x}^2} = \frac{\partial D'^T(\bar{x})}{\partial \bar{x}};, \hat{\mathbf{x}} = -\frac{\partial^2 D}{\partial \mathbf{x}^2}^{-1}\frac{\partial D}{\partial \mathbf{x}}$ ind. for x=0, use $\hat{\mathbf{x}} > 0.5$ 2. Contrast if $|D(\hat{\mathbf{x}})| = |D + \frac{1}{2}\frac{\partial D^T}{\partial \mathbf{x}}\hat{\mathbf{x}}| < 0.03$, reject. 2.2 *edgeThres* pca frm 2x2 Hessian matrix, eigenvalues of H pca ratio of D; $\frac{\operatorname{Tr}(\mathbf{H})^2}{\operatorname{Det}(\mathbf{H})} = \frac{(\alpha+\beta)^2}{\alpha\beta} = \frac{(r+1)^2}{r}$, r<10 is accepted ③*Orientation Assignment* Orientation histogram, 36 bins, 4x4 size, weighted by gradient magnitude and gaussian-weighted circular window, σ =1.5*scale of keypoint, highest peak in the histogram is taken and any peak above 80% also, calculate the orientation. $>$ 1 keypoints with same location and scale, but different directions=more stability ④*Descriptor* gradient(keypoint) mθ, rotated w.r.t keypoint orientation, weighted - guass, smoothly-avoid sudden changes in desc.. ;normalized to unit length=descriptor invariant- affine changes in illumination, non-linear illumination-3D surfaces- differing orientations- change relative magnitudes for some gradients - thresholding the values in the unit feature vector to each be no larger than 0.2(experiment)-imp to distribution than value, and then renormalizing to unit length-16x16 around keypoint - 16 sub-blks(8 bins each) of 4x4 = 128bin - feature descriptor of the keypoint ⑤*Matching Homography* - to get translation into, $\bar{x}' = \mathbb{H}\bar{x}$ 1. Euclidian-rottrans 2. Similarity-scaling 3. Affine - shearing, 4.Projective - collinearity(aBc)-violated in differnt plane, concurrency (abcd) $\left[\begin{array}{ccccccccc} x & y & 1 & 0 & 0 & 0 & -x'x & -x'y & -x^1 \\ 0 & 0 & 0 & x & y & 1 & -y'x & -y'y & -y' \end{array}\right][h] = 0;\ A = UDV^\top \rightarrow \bar{h}$ last row of $V^\top$; $\bar{q} = V^\top\bar{p}$; $\mathbb{D}\vec{q} = \left[\begin{array}{cc} \alpha_1 & 0 \\ 0 & \alpha_2 \end{array}\right]\left[\begin{array}{c} q_x \\ q_y \end{array}\right] = \left[\begin{array}{c} 0 \\ 0 \end{array}\right]$ $\alpha_1 \geq \alpha_2$; $\min|A\vec{p}| = \min|D\vec{q}|$; $\bar{q} = [0, 0, \ldots, 1]^\top$; $\bar{p} = V\bar{q}$; ⑥*RANSAC* 1. Select n¿5 matched pairs (dist of 1st/2nd¡0.8-rejected eliminates 90% false matches) 2. Compute H 3. Inliers $d\left(x'_{in}, Hx_{in}\right) < \varepsilon$ 4. Max inliers - prob, 5. Recompute H with all inliers ⑥*Least Square* $\mathbb{H}^* = \underset{H}{\operatorname{argmin}} \sum_n \left(x'_n - Hx_n\right)^2$;Grad descent $h_{11}^{\text{new}} = h_{11}^{\text{old}} - \eta \left.\frac{\partial E(H)}{\partial h_{11}}\right|_{h_{11} = h_{11}^{\text{ord}}}; -2\sum_n^1 \left(x'_n - h_{11}x_n - h_{12}y_n - h_{13}\right)x_n$

4. Camera Parameters $\bar{x} = \mathbb{P}\bar{X}$, both are in homogeneous coord. $\bar{X}_c = \mathbb{TR}\bar{X}_w$Xc and Xw just have translation and then rotation - Rotate and translates, otherwise translation will also rotate.$\mathbb{R} = \left[\begin{array}{ccc} 1 & 0 & x_0 \\ 0 & 1 & y_0 \\ 0 & 0 & 1 \end{array}\right]\left[\begin{array}{ccc} f_x & 0 & 0 \\ 0 & f_y & 0 \\ 0 & 0 & 1 \end{array}\right]$, K - scale down(lens), xo,yo - translate from centre of image to top left, every image is at Z=1 plane normal to the Z axis. Usually fx = fy (focal length). $\mathbb{TR} = \left[\begin{array}{cc} \mathbb{R} & \bar{t} \\ 0 & 1 \end{array}\right]$; $[\mathbb{R}|\vec{t}]\bar{X}_w(3X4, 4X1)$; $\bar{x} = \mathbb{P}\bar{X}_w = \mathbb{K}\bar{X}_c = \mathbb{K}[\mathbb{R}|\bar{t}]\bar{X}_w$ K - intrinsic to the camera (calibration etc.), R,t - extrinsic i.e depends on the position of camera etc. ① *Forward propagation* - $\bar{x} = \mathbb{P}\bar{X}$; don't forget to normalise $\bar{x} = \bar{x}./x(\bar{3})$ ② *Backward propagation* Given a point in image, it corresponds to a ray in real world. Ray passing through camera *center* $\bar{C} = -\mathbb{R}^{-1}\bar{t}$ ($[\mathbb{R}[\bar{t}]\bar{C} = \mathbb{R}\bar{C} + \bar{t}, \mathbb{R}$ is orthonormal)and $\hat{X}_w = \mathbb{P}^+\bar{x} = \left(\mathbb{P}^\top\mathbb{P}\right)^{-1}\mathbb{P}^\top\bar{x}$, P is a 3x4 matrix. Hence $\bar{X} = \bar{c} + \lambda\mathbb{P}^+\bar{x}$. **Callibration** Given $\bar{x}$&$\bar{X}$-N pairs, estimate K,R,t.→① *Compute P* $\bar{x} = \mathbb{P}\bar{X}_w; x_i\left(p_{31}x_\omega + p_{32}y_\omega + p_{33}z_\omega + p_{34}\right) = p_{11}x_\omega + p_{12}y_\omega + p_{13}z_{\omega+}p_{14}; \mathbb{A}_D = \bar{0}$; where, $\bar{p} = \left[p_{11}p_{12}p_{13}p_{44}p_{21}p_{21}p_{33}p_{29}p_{31}p_{32}p_{33}p_{34}\right]^\top$; 1 pair - 2 equations → atleast 6 pairs, 1. Using SVD, $\mathbb{A} = \mathbb{UDV}^\mathbb{T}, \bar{p}$ is the last row of $\mathbb{V}^\mathbb{T}$, for $\mathbb{A}\bar{p} = 0$ 2. Further use MLE - Using Newtons method → $x^{\text{new}} = x^{\text{old}} - \alpha\nabla E(x)/\nabla\nabla E(x) = x^{\text{old}} - \alpha\mathbb{H}^{-1}(x)\nabla E(x)$; $P_{11}^{\text{new}} = P_{11}^{\text{old}} - \alpha\sum_n\left(x_n - P_{11}^{\text{old}}X_n - P_{12}Y_n - P_{13}Z_n - P_{14}\right)X_n / -\sum_n X_n^2$; $E(\mathbb{P}) = \sum_n\left(\left[\begin{array}{c} x_n \\ y_n \\ 1 \end{array}\right] - \mathbb{P}\left[\begin{array}{c} x_0 \\ y_n \\ z_n \end{array}\right]\right)^2$, Hessian = $Hf(x,y) \equiv \left|\begin{array}{cc} \frac{\partial^2 f}{\partial x^2} & \frac{\partial^2 f}{\partial x^2 y} \\ \frac{\partial^2 f}{\partial y \partial x} & \frac{\partial^2 f}{\partial y^2} \end{array}\right|$, Better - Levenberg method - $x^{\text{new}} = x^{\text{old}} - (\mathbb{H}(x) + \mu\mathbb{I})^{-1}\nabla E(x)$ ②*Decompose P to KRt* Extract $\mathbb{M}$ from P, M is a 3x3 submatrix of P; Decompose M to KR, using RQ method; and $\bar{t} = \mathbb{K}^{-1}\left(\begin{array}{c} P_{14} \\ P_{24} \\ P_{34} \end{array}\right)$, non homogeneous t; Skew - $\left[\begin{array}{ccc} 1 & s & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{array}\right]$, most cameras s =0, vertical skew is rare. Checkered box - pairs, find corresponding pairs.

5. Depth from Stereo Stereo - horizontally seperated, left and right eye; $\bar{x}'^T\mathbb{F}\bar{x} = 0; l' = \mathbb{F}\bar{x}$ F is a 3x3 matrix with rank 2. Two points create a line through cross product, cross product to matrix is skew symmetric matrix $\left[e^T\right]_X = \left[\begin{array}{ccc} 0 & -e'_z & e'_y \\ e'_z & 0 & -e'_x \\ -e_y & e'_x & 0 \end{array}\right]$; $\bar{l}' = [e']_x\mathbb{H}\bar{x} = \mathbb{F}\bar{x}$; Now let's derive F from Ps, from previous we know $\mathbb{P}^+\bar{x}$ and C for a ray, let's transform the ray into line in image, $\mathbb{P}^+\bar{x} \rightarrow \mathbb{P}'\mathbb{P}^+\bar{x}$ and $\bar{C} \rightarrow \mathbb{P}'\bar{C} = \bar{e}'$; Hence, $\bar{l}' = (\mathbb{P}'\bar{c}) \times (\mathbb{P}'\mathbb{P}^+\bar{x}) = [\bar{e}']_x\mathbb{P}'\mathbb{P}^+\bar{x} = \mathbb{F}\bar{x}$, hence F is independent of the world or the scene. $\mathrm{l} = \mathrm{F}^\top\mathrm{x}'$ is the epipolar line corresponding to x'. $\mathrm{Fe} = \mathbf{0}, \mathrm{F}^\top\mathbf{e}' = \mathbf{0}$ ⋄ Canonical cameras, $\mathrm{P} = [\mathrm{I}|\mathrm{o}], \mathrm{P}' = [\mathrm{M}|\mathrm{m}]$ $\mathrm{F} = [\mathrm{e}']_\times \mathrm{M} = \mathrm{M}^{-\top}[\mathrm{e}]_\times$, where $\mathrm{e}' = \mathrm{m}$ and $\mathrm{e} = \mathrm{M}^{-1}\mathrm{m}$; Rectified image - search along the row.

6. Markov Random Field - graph expresses conditional dependence structure between rand vars. *Bayes Theorem* - $p(x,y) = p(x|y)p(y)$; If x,y are independent, $p(x|y) = p(x)$; $p(x|d) = \frac{p(d|x)p(x)}{p(d)} \rightarrow$ posterior $= \frac{\text{likelihood} \times \text{prior}}{\text{evidence}}$; d-obs. data, x -rand var. $p(d) = \int p(d|x)p(x)dx$, in most cases is intractable. ⋄ 3 Probabilistic Inference methods - ① MLE - max likelihood est. $x^* = \operatorname{argmax} p(d|x)$, return x that maximises $p(d|x)$, Weakness - treats every candidate equally ② MAP - max a posterior - $x^* = \operatorname{argmax} p(x|d) = \operatorname{argmax} p(x|d)p(x)$, since p(d) is indep. of x, can be ignored in argmax. ③ Full baysian - not max but direct $p(x|d)$ *Factorisation* - $p\left(x_1, x_2, x_3\right) = p\left(x_3|x_2, x_1\right)p\left(x_2|x_1\right)p\left(x_1\right) = p\left(x_3|x_2\right)p\left(x_2|x_1\right)p\left(x_1\right)$, if x3 is indep. of x1. Factorisation reduces the complexity of joint prob. computation. ① Directed - $p\left(x_1, \ldots, x_N\right) = \prod_n p\left(x_n|x_{pa}\left(x_n\right)\right); x_{pa}$ are the parent for xn. ② Undirected - $p\left(x_1 \ldots x_N\right) = \frac{1}{z}\prod_c \Phi_c\left[x_1 \ldots x_N\right]$ c -set of cliques. $p(\{x\}, \{d\}) = \frac{1}{z}\prod_i \phi\left[x_i, d_i\right]\prod_{j \in N_i}\phi\left[x_i, x_j\right]$, to make i minium, we log the whole and $\{x\}^* = \operatorname{argmin}\sum_i f_d\left(x_i, d_i\right) + \sum_{j \in N_i} f_p\left(x_i, x_j\right)$ ⋄ *Graphcut* - to implement argmin, the cost of cut = total cost of assignment - data and prior terms sum. The cut is the assignment i.e if x1 B are but, x1 is assigned to B. $\diamond$ *Belief Propogation* - belief $= b_i(x_i) = k\phi_i(x_i)\prod_{j\in N_i} m_{ji}(x_i)$; messeage $= m_{ij}(x_j) = \sum_{x_i}\phi i(x_i)\psi_{ij}(x_i,x_j)\prod_{k\in N_i/j} m_{ki}(x_i)$

7. Depth from Video $D^* = \operatorname{argmin}\sum_x\left(fd(x_0,D(x),I,I') + \lambda\sum_{y\in N_x} f_P(D(x),D(y))\right)$, $f_d(x,D(x),I,I') = [I(x) - I'(x+D(x))]$ $f_P(D(x),D(y)) = (D(x)-D(y))^2$, fp can be made into a matrix. *Unrectified Image* - $x' \sim \mathbb{K}'\mathbb{R}'\mathbb{R}^\top\mathbb{K}^{-1}x + d\mathbb{K}'\mathbb{R}'(\bar{c}-\bar{c}')$, $\rightarrow$ derivation $x^h = \mathbb{K}[\mathbb{R}|\tilde{t}]X_\infty = \mathbb{K}[\mathbb{R}|\bar{t}]\begin{bmatrix}\hat{X}_\infty \\ 0\end{bmatrix} = \mathbb{K}\mathbb{R}\hat{X}_\infty$ - *3x3*, hence $\hat{X}_\infty = \mathbb{R}^\top\mathbb{K}^{-1}x^h$, is not homo. $x^h_\infty = \mathbb{P}'X_\infty = \mathbb{K}'[\mathbb{R}'|\bar{t}']\begin{bmatrix}\hat{X}_\infty \\ 0\end{bmatrix} = \mathbb{K}'\mathbb{R}'\hat{X}_\infty = \mathbb{K}'\mathbb{R}'\mathbb{R}^\top\mathbb{K}^{-1}x^h$; $\vec{e}_{t'} = \mathbb{P}'\bar{c}_t = \mathbb{K}'[\mathbb{R}'|\bar{t}']\bar{c}_t = \mathbb{K}'\mathbb{R}'\left[\mathbb{I}\,\middle|\,[\mathbb{R}']^{-1}\bar{t}'\right]\bar{c}_t = \mathbb{K}'\mathbb{R}'\left[\bar{C}_t - \bar{C}'_t\right]$; $\bar{c}_{t'} = -(\mathbb{R}')^{-1}\bar{t}'$, here C is 3x1 as matrix is converted to addition. $x'^h \sim x'^h_\infty + d\bar{e}_{t'}$, x is in homography cord. If 3 images, $d_{12} = d_{13}$. $d_i(x,y) = \frac{f_i b}{z_i(x,y)}$ long baseline better - b magnifies, however they suffer from occlusions. *Depth from Video - paper* effectively suppress temporal outliers by use of stat. info from multiple frames. to improve the final recon. qual., used optical flow-find corres. pxls in the subsequent frames of same camera, and enforced temporal consistency in reconstructing successive frames.depth error in conv. stereo methods grows quad with depth, G et al.- multibaseline and multiresolution stereo method-achieve constant depth accuracy-varying baseline and resolution proportionally to depth., maintaining the temporal coherence, surprisingly consistent and accurate dense depth maps obtained. 1 *Structure from Motion* 2 *Disparity Initialisation* $L_{init}(\mathbf{x},D_t(\mathbf{x})) = \sum_{t'} p_c(\mathbf{x},D_t(\mathbf{x}),I_t,I_{t'})$; $p_c(\mathbf{x},d,I_t,I_{t'}) = \sigma_c/\sigma_c + \|I_t(\mathbf{x}) - I_{t'}(l_{t,t'}(\mathbf{x},d))\|$; p_c measures the color similarity. $E^t_d\left(D_t;\hat{I}\right) = \sum_{\mathbf{x}} 1 - u(\mathbf{x})\cdot L_{\text{init}}(\mathbf{x},D_t(\mathbf{x}))$; $u(\mathbf{x}) = 1/\max_{D_t(\mathbf{x})} L_{init}(\mathbf{x},D_t(\mathbf{x}))$ - adaptive norm. - imposing stronger smoothness constraint in the flat regions than in the textured ones. $E_s(D_t) = \sum_{\mathbf{x}}\sum_{\mathbf{y}\in N(\mathbf{x})}\lambda(\mathbf{x},\mathbf{y})\cdot\rho(D_t(\mathbf{x}),D_t(\mathbf{y}))$; preserve discontinuity, $\lambda(\mathbf{x},\mathbf{y})$ defined in anisotropic way, encouraging the disparity discon. to be coincident with abrupt intensity/color change. $\lambda(\mathbf{x},\mathbf{y}) = w_s\cdot\frac{u_\lambda(\mathbf{x})}{\|I_t(\mathbf{x}) - I_t(\mathbf{y})\|+\varepsilon}$; $u_\lambda(\mathbf{x}) = |N(\mathbf{x})|/\sum_{\mathbf{y}'\in N(\mathbf{x})}\frac{1}{\|I_t(\mathbf{x}) - I_t(\mathbf{y}')\|+\varepsilon}$; ws denotes the smoothness strength and e controls the contrast sensitivity. adap. smooth. term imposes smoothness in flat regions while preserving edges in textured ones. 3 *Bundle* $L(\mathbf{x},d) = \sum_{t'} p_c(\mathbf{x},d,I_t,I_{t'})\cdot p_v(\mathbf{x},d,D_{t'})$; $p_v(\mathbf{x},d,D_t) = \exp\left(-\frac{\|\mathbf{x} - l_t(\mathbf{x}',D_t(\mathbf{x}'))\|^2}{2\sigma_d^2}\right)$; $E^t_{init}\left(D_t;\hat{I}\right) = \sum_{\mathbf{x}}[1 - u(\mathbf{x})\cdot L_{init}(\mathbf{x},D_t(\mathbf{x})) + \sum_{\mathbf{y}\in N(\mathbf{x})}\lambda(\mathbf{x},\mathbf{y})\cdot\rho(D_t(\mathbf{x}),D_t(\mathbf{y}))]$Final data term - $E_d\left(D_t;\hat{I},\hat{D}/D_t\right) = \sum_{\mathbf{x}} 1 - u(\mathbf{x})\cdot L(\mathbf{x},D_t(\mathbf{x}))$ 4 *Space time fusion*

8. Projective 3D reconstruction Given $\bar{x}_n \& \bar{x}'_n$, estimate $\mathbb{P},\mathbb{P}'$ and $\bar{X}_n$. Without knowing P, the solution is not unique - projective reconstruction. Projective - preserves lines to lines, Afffine - preserves parallel lines, Metric - preserves angle. STEPS $\rightarrow$ 1 *Compute F* $\bar{x}'^T\mathbb{F}\bar{x} = 0$;$x'xf_{11} + x'yf_{12} + x'f_{13} + y'xf_{21} + y'yf_{22} + y'f_{23} + xf_{11} + yf_{32} + f_{33} = 0$; $[\ x'x\ \ x'y\ \ x'\ \ y'x\ \ y'y\ \ y'\ \ x\ \ y\ \ 1\ ]$; $\mathbb{A}f = 0$; - SVD with atleast 8 pairs. 2 *Factor F* After obtaining F(using SVD on A), again we do SVD on F(diff). assume $\mathbb{P} = \mathbb{K}[\mathbb{I}|0]$; $\mathbb{P}' = \mathbb{K}'[\mathbb{R}|\bar{t}]$ - world cord. coincides with first camera. $\mathbb{F} = [\mathbb{P}'\bar{c}]_x\mathbb{P}'\mathbb{P}^+ = [\mathbb{K}'[\mathbb{R}|\bar{t}]\bar{c}]_x\mathbb{K}'[\mathbb{R}|\bar{t}][\mathbb{K}[\mathbb{I}|0]]^+$; $\mathbb{P}'\bar{c} = \mathbb{K}'[\mathbb{R}|\bar{t}][0001]^T = \mathbb{K}'\bar{t}$; $\mathbb{P}^+ = (\mathbb{K}[\mathbb{I}|0])^+ = [\mathbb{K}|0]^+ = \left[\frac{\mathbb{K}^{-1}}{0^\top}\right]$; $\mathbb{F} = [\mathbb{K}'\bar{t}]_x\mathbb{K}'[\mathbb{R}|t]\begin{bmatrix}\mathbb{K}^{-1} \\ 0^T\end{bmatrix} = [\mathbb{K}'\bar{t}]_x\mathbb{K}'\mathbb{R}\mathbb{K}^{-1} = [\mathbb{U}\mathbb{Z}\mathbb{U}^\top][\mathbb{U}\mathbb{Z}^\top\mathbb{D}\mathbb{V}^\top] = [t]_x\mathbb{M}$, d33 of *mathbbD* = 0, as F is rank 2. $[\bar{z}]_x A = A^*\left[A^{-1}\bar{z}\right]_x = A^{-T}\left[A^{-1}\bar{z}\right]_x$, where$A^* = \det(A)A^{-T}$; $[\mathbb{K}'\vec{t}]_x K' = [\bar{q}]_x\mathbb{K}' = (\mathbb{K}')^{-\top}\left[(\mathbb{K}')^{-1}\bar{q}\right]_x = (K')^{-T}[\bar{t}]_x$; $E = [\bar{t}]_x\mathbb{R} = \mathbb{S}\mathbb{R} = (\mathbb{U}\mathbb{Z}\mathbb{U}^\top)(\mathbb{U}\mathbb{W}\mathbb{V}^\top) = \mathbb{U}\mathbb{Z}\mathbb{W}\mathbb{V}^\top$, Essential matrix is similar to F, with known F, normalised x,y coordinates. Where, $\mathbb{Z} = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0\end{bmatrix}$ and $W = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1\end{bmatrix}$, and $\mathbb{Z}\mathbb{W} = \operatorname{diag}(1,1,0)$ 3 *Compute* $\mathbb{P}\mathbb{P} = [\mathbb{I}|0]$; $\mathbb{P}' = [\mathbb{M}|t]$ 4 *Estimate Xn* After representing x,y in terms of X,Y,Z, we get $(x[p_{31},p_{32},p_{33},p_{34}] - [p_{11},p_{12},p_{13},p_{14}])[XYZ1]^T = 0$, hence $[xP_3^T - P_1^T; yP_3^T - P_2^T; x'P_3^T - P_1^T; y'P_3^T - P_2^T]^T\bar{X} = 0$, $\mathbb{A}\bar{X} = 0$; note that we have 3 unknowns but 4 equations for every pair of points. But there is *ambiguity* $\bar{x} = \mathbb{P}\bar{X} = \mathbb{P}\left(\mathbb{H}\mathbb{H}^{-1}\right)\bar{X} = (\mathbb{P}\mathbb{H})(\mathbb{H}^{-1}\bar{X}) = \tilde{\mathbb{P}}\tilde{X}$; 1 Projective to metric recon - compute H s.t $\bar{X}_{E,n} = H\bar{X}_n$, Xn - computed and XEn is the real world points. - use 5 or more points (3 eq. per point), $\mathbb{P}_M = \mathbb{P}\mathbb{H}^{-1}$; $\bar{X}_{M,n} = \mathbb{H}\bar{X}_n$; $\mathbb{P}'_M = \mathbb{P}'\mathbb{H}^{-1}$; 2Structure from motion - Multiview geometry *SLAM-Simultaneous Localisation and mapping* M imgs of N 3D points $\bar{X}_n$ (struction) and P_m (motion)=Rm, tm, since K is known(self calb.); $\min_{\mathbb{P}_m,\bar{X}_n}\sum_m\sum_n d\left(\mathbb{P}_m\bar{X}_n, x_{mn}\right)^2$ Steps- 1. Initialise R,t and structure Xn from 2 img (K is known, self calib.) [2. Compute $\mathbb{P}_{m+1}$ using all the known 3D points (computed in 1) and visible in image. 3. Refine 3D points using triangulation, compute new 3D points] -repeat 4 frames.

9. Optical flow $\neq$ Motion flow. *Brightness Constancy Constraint* $I(x,y,t) = I(x+u,y+v,t+1)$; $\frac{dI(x,y,t)}{dt} = \frac{dI}{dx}\frac{dx}{dt} + \frac{dI}{dy}\frac{dy}{dt} + \frac{dI}{dt} = 0$; $I_xu + I_yv + I_t = 0$; Lucas Kanhade obj. func. $\min E(u,v) = \min(I_xu + I_yv + I_t)^2 \rightarrow$ dE/du,dE/dv - $\begin{bmatrix} I_xI_x & I_xI_y \\ I_yI_x & I_yI_y\end{bmatrix}\begin{bmatrix} u \\ v\end{bmatrix} = -\begin{bmatrix} I_xI_t \\ I_yI_t\end{bmatrix}$, but det(A) = 0, not invertible, hence we take a patch(same flow inside patch) for the energy function, repeating it $\sum_{i,j} Ix_{i,j}{}^2$, even here if the patch doesn't contain texture(edges, corners), then det(A) = 0; So we use guassian kernel over a patch, $(G * I_x^2)(G * I_y^2) \neq (G * I_xI_y)^2$; *Horn Schunk OF-Dense(every pixel) and global* $E(u,v) = (I_xu + I_yv + I_t)^2 + \alpha^2\left(\|\nabla u\|_2^2 + \|\nabla v\|_2^2\right)$;$\nabla u(x,y) = \left(\frac{\partial u}{\partial x},\frac{\partial u}{\partial y}\right)^T = (u_x,u_y)^T$;$\|\nabla u\|_2 = \sqrt{u_x^2 + u_y^2}$; Global obj. function $\int E(u,v)dxdy = 0$, we need to find u(x,y) for every pixel. This can be solved using *Euler Lagrange eq.* If $J = \int F(t,\bar{y},\bar{y}')\,dt$, J will have a stationary point if $\frac{\partial F}{\partial y} - \frac{\partial}{\partial t}\left(\frac{\partial F}{\partial y'}\right) = 0$; Using this, $\frac{\partial E}{\partial u} - \frac{\partial}{\partial x}\frac{\partial E}{\partial U_x} - \frac{\partial}{\partial y}\frac{\partial E}{\partial U_y} = 0$; $\frac{\partial}{\partial x}\left(\frac{\partial E}{\partial u_x}\right) = \frac{\partial}{\partial x}\left(\frac{\partial\alpha^2\left[\sqrt{u_x^2+u_y^2}\right]^2}{\partial u_x}\right) = \frac{\partial}{\partial x}(2\alpha^2U_x) = 2\alpha^2U_{xx}$;$\triangle U = U_{xx} + U_{yy}$, hence $2(I_xu + I_yv + I_t)I_x - 2\alpha^2\Delta u = 0$; $2(I_xu + I_yv + I_t)I_y - 2\alpha^2\Delta v = 0$; $U_{xx} = (u_{i-1} + u_{i+1}) - 2u_i$; Let $\Delta U = \bar{u} - u$, $\bar{u}$ is average at i,i+1. $\begin{bmatrix}\alpha^2 + I_x^2 & I_xI_y \\ I_xI_y & \alpha^2 + I_y^2\end{bmatrix}[uv]^T = \begin{bmatrix}\alpha^2\bar{u} - I_xI_t \\ \alpha^2\bar{v} - I_yI_t\end{bmatrix}$; $\mathbb{A}\bar{\omega} = \bar{b}$;(2Nx2N).(2Nx1)=(2Nx1); This can be expanded for N pixel(i,j), $\begin{bmatrix}\alpha^2 + I_{x_{ij}}^2 & I_{x_{ij}}I_{y_{ij}} \\ I_{x_{ij}}I_{y_{ij}} & \alpha^2 + I_{y_{ij}}^2\end{bmatrix}\begin{bmatrix}u_{ij} \\ v_{ij}\end{bmatrix} = \begin{bmatrix}\alpha^2\bar{u}_{ij} - I_{x_{ij}}I_{t_{ij}} \\ \alpha^2\bar{v}_{ij} - I_{y_{ij}}I_{t_{ij}}\end{bmatrix}$;here $det(\mathbb{A}) = \alpha^2\left(\alpha^2 + I_{x_{ij}}^2 + I_{y_{ij}}^2\right)$; $(\alpha^2 + I_{x_{ij}}^2 + I_{yij}^2)\begin{bmatrix}u_{ij} - \bar{u}_{ij} \\ v_{ij} - \bar{v}_{ij}\end{bmatrix} = \begin{bmatrix}-I_{x_{ij}}(I_{x_{ij}}\bar{u}_{ij} + I_{yij}\bar{v}_{ij} + I_{t_{ij}}) \\ -I_{y_{ij}}(I_{x_{ij}}\bar{u}_{ij} + I_{yij}\bar{v}_{ij} + I_{t_{ij}}\end{bmatrix}$; $u^{new}_{ij} = \bar{u}^{old}_{ij} - \ldots$. Since $A = \begin{pmatrix} a & b \\ c & d\end{pmatrix}$; $\mathrm{A}^{-1} = \frac{1}{\det(A)}\begin{pmatrix} d & -b \\ -c & a\end{pmatrix}$ *Robust* - Coarse to fine(original HS doesn't work for large disp or motion), easy to init u_{ij}, interpolate, median, GCC($\nabla I(x,y,t) = \nabla I(x+u,y+v,t+1)$; $E = E + \delta\left\|\begin{matrix} I_{xx}u + I_{xy}v_fI_{xt} \\ I_{yx}u + I_{yy}v_fI_{yt}\end{matrix}\right\|^2\begin{bmatrix}\frac{\partial I}{\partial x}(x,y,t) \\ \frac{\partial}{\partial y}(x,y,t)\end{bmatrix} = \begin{bmatrix}\frac{\partial I}{\partial x}(x+u,y+v,t+1) \\ \frac{\partial I}{\partial y}(x+u,y+v,t+1)\end{bmatrix}$) ; 1. Robust obj. function 2. Coarse to fine 3. Interpolation 4. Median Filtering 5. Pre processing 6. GCC; when u suffers from noise, penalty is very high, hence trunkte it, more robust functions - Lorents($\log\left(1 + \frac{1}{2}\left(\frac{x}{\sigma}\right)^2\right)$); Charbonnier$f(x) = (x^2 + \varepsilon^2)^a$; 1.Img pyramid2. OF at coarse 3.warp up

Structure decomp. Texture is indep from light changes.$\min_{I_S}\sum_{\bar{x}}(I_s(\bar{x}) - I(\bar{x}))^2 + \lambda|\nabla I_S(\bar{x})|_2$, Global cost $J = \iint E(I_s)\,dxdy$; using Euler-Lag. $\nabla J = \frac{\partial E}{\partial I_S} - \frac{\partial}{\partial x}\frac{\partial E}{\partial I_{Sx}} - \frac{\partial}{\partial y}\frac{\partial E}{\partial I_{Sy}} = 0$; $\frac{\partial E}{\partial T_S} = 2(I_S(\bar{x}) - I(\bar{x}))$;$\frac{\partial E}{\partial I_{s_x}} = \frac{\lambda}{2}\left(\frac{2I_{sx}}{\sqrt{I_{sx}^2 + I_{sy}^2}}\right)$;$I_\delta^{new} = I_s^{old} - \alpha\nabla J(I_s)|_{I_s = I_s^{old}}$, L2 norm generates blurry edge. 0-0 vs $\square - 0$; $If(I_{sx},I_{sy})$ are mostly zero (sparse) r the values of Is is sharp. L1