Statistical Multilinear Models:

MICA
Multilinear Projection

PART I: 2D Vision

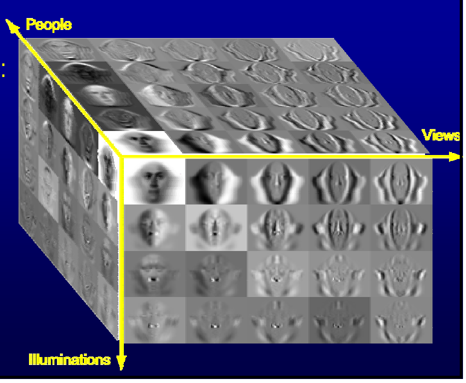
- Appearance-Based Methods
  - **Statistical Linear Models:**
    - **PCA**
    - ICA, FLD
    - Non-negative Matrix Factorization, Sparse Matrix Factorization
  - **Statistical Tensor Models:**
    - Multilinear PCA,
    - Multilinear ICA
    - Multilinear Projection

Today ➔ { Multilinear PCA, Multilinear ICA, Multilinear Projection

---

PCA:

TensorFaces:



Strategic Data Compression = Perceptual Quality

- TensorFaces data reduction in illumination space primarily degrades illumination effects (cast shadows, highlights)
- PCA has *lower mean square error* but *higher perceptual error*

Original	TensorFaces	TensorFaces	PCA
176 basis vectors	6 illum + 11 people param. 66 basis vectors	3 illum + 11 people param. 33 basis vectors	33 parameters 33 basis vectors

Dimensionality Reduction - Truncation

$$\left\| \mathcal{D} - \hat{\mathcal{D}} \right\|^2 \leq \sum_{i_1=R_1}^{I_1} \sigma_{i_1}^2 + \sum_{i_2=R_2}^{I_2} \sigma_{i_2}^2 \cdots + \sum_{i_N=R_N}^{I_N} \sigma_{i_N}^2$$

Iterative Multilinear Model - Data Reduction
(CVPR'03)

$$\mathbf{E} = \left\| \mathcal{D} - \mathcal{Z} \times_1 \mathbf{U}_1 \times \ldots \times_n \mathbf{U}_n \times \ldots \times_N \mathbf{U}_N \right\| + \sum_{n=1}^{N} \lambda_n \left\| \mathbf{U}_n \mathbf{U}_n^T - \mathbf{I} \right\|$$

- Iterative data reduction approach:
  - Optimize mode per mode in an iterative way
  - Alternating Least Squares [Golub & Van Loan] improves data fit

Iterative Multilinear Model

1. Initialize $U_1^0, U_2^0, \ldots, U_N^0$:
   - Compute $U_1, U_2, \ldots, U_N$ using N-Mode SVD and
   - Truncate each mode matrix
2. Iterate:
   - $\mathcal{U}_1^t = \mathcal{D} \times_2 U_2^{t-1^T} \times_3 \cdots \times_N U_N^{t-1^T}$

     $U_1^t = \text{svd}(\mathcal{U}_{1(1)}^t)$
   - $\mathcal{U}_2^t = \mathcal{D} \times_1 U_1^{t^T} \times_3 U_3^{t-1} \cdots \times_N U_N^{t-1^T}$

     $U_2^t = \text{svd}(\mathcal{U}_{2(2)}^t)$
3. $\mathcal{Z} = \mathcal{D} \times_1 U_1^{t^T} \times_2 U_2^{t^T} \times_3 \cdots \times_N U_N^{t^T}$

Person Specific TensorFaces

$$\mathcal{B} = \mathcal{Z} \times_1 U_{people} \times_5 U_{pixels}$$

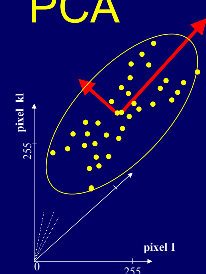
- Basis spanning one person's set of images

Perspective on Our Face Recognition Approach

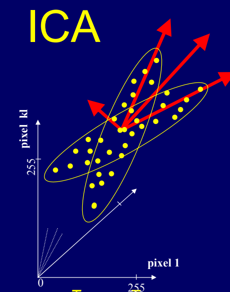
	Linear Models	Our Nonlinear (Multilinear) Models
2nd -Order Statistics (covariance)	**PCA** Eigenfaces	**Multilinear PCA** TensorFaces
Higher -Order Statistics	**ICA**	**Multilinear ICA** Independent TensorFaces

Vasilescu & Terzopoulos, Learning 2004

PCA / ICA



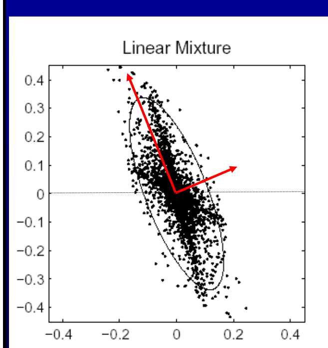
$D = USV^T$ (basis matrix: U; coefficient matrix: SV^T)



$$D = U\,W^T\,W^{-T}\,SV^T = K\,C$$

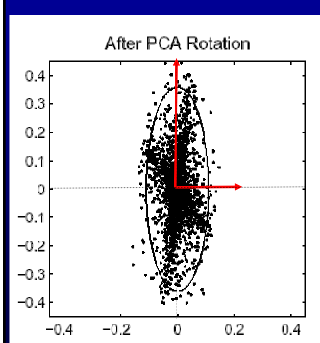
(independent components: K; coefficient matrix: C)

Geometric View of ICA



$$D = USV^T$$

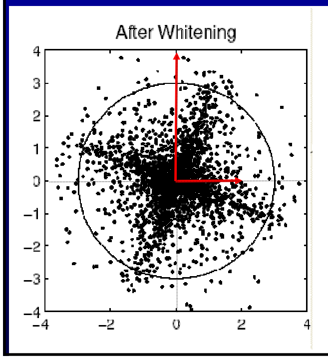
Geometric View of ICA



$$D = USV^T$$

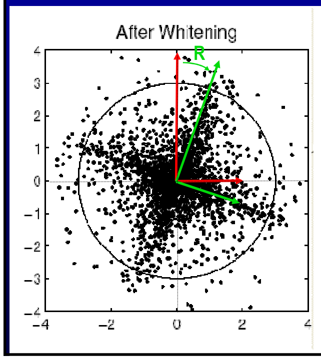
$$D' = U^T D$$

Geometric View of ICA



$$D = USV^T$$
$$D' = U^T D$$
$$D'' = S^{-\frac{1}{2}} U^T D$$

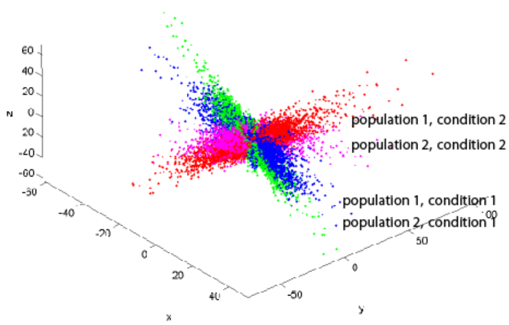
Geometric View of ICA



$$D = USV^T$$
$$D' = U^T D$$
$$D'' = S^{-\frac{1}{2}} U^T D$$
$$D''' = RS^{-\frac{1}{2}} U^T D$$
$$D = \underbrace{US^{\frac{1}{2}}R^T}\underbrace{RS^{-\frac{1}{2}}}SV^T$$
$$D = \underbrace{UW^{-1}}WSV^T$$

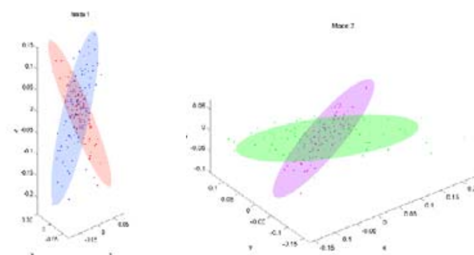
Independent Components

Toy Example: Observed Data



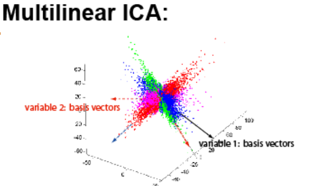
Toy Example: Hidden Variables

- These variables were used to create the observed data

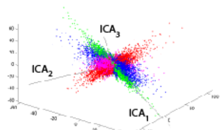


Basis Vectors

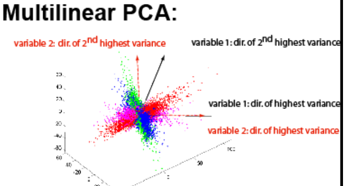
Multilinear ICA:



ICA

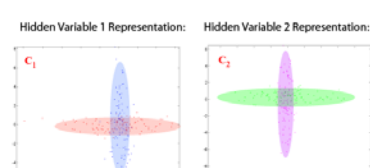


Multilinear PCA:

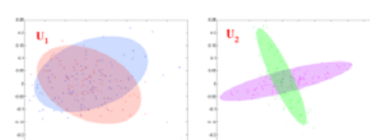


Extracted Hidden Variables

- MICA



- MPCA



N-Mode ICA

1. For n=1,…,N, compute matrix $\mathbf{U}_n$ by computing the SVD of the flattened matrix $\mathbf{D}_{(n)}$ and setting $\mathbf{U}_n$ to be the left matrix of the SVD. Compute $\mathbf{W}_n^T$ using ICA. Our new mode matrix is $\mathbf{K}_n$

$$\mathbf{D}_{(n)} = \mathbf{U}_n \mathbf{Z}_{(n)} \mathbf{V}_n^T = \underbrace{\left(\mathbf{U}_n \mathbf{W}_n^T\right)}\mathbf{W}_n^{-T} \mathbf{Z}_{(n)} \mathbf{V}_n^T$$

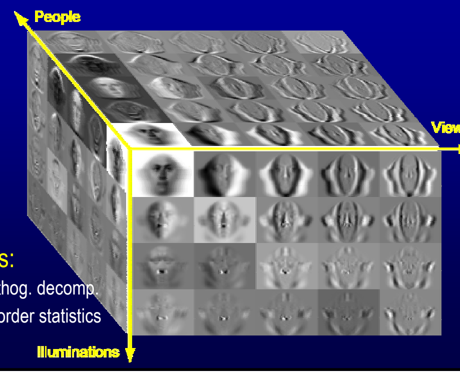
$$= \mathbf{K}_n \quad \mathbf{W}_n^{-T} \mathbf{Z}_{(n)} \mathbf{V}_n^T$$

2. Solve for the core tensor as follows

$$\mathcal{S} = \mathcal{D} \times_1 \mathbf{K}_1^{-1} \times_2 \mathbf{K}_2^{-1} \times \cdots \times_n \mathbf{K}_n^{-1} \times \cdots \times_N \mathbf{K}_N^{-1}$$

$$\mathcal{S} = \mathcal{Z} \times_1 \mathbf{W}_1^{-T} \times_2 \mathbf{W}_2^{-T} \times \cdots \times_n \mathbf{W}_n^{-T} \times \cdots \times_N \mathbf{W}_N^{-T}$$

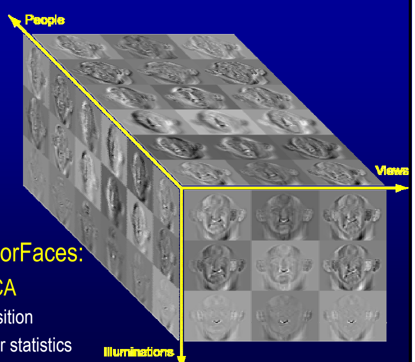
PCA:



TensorFaces:

- Multilinear orthog. decomp.
- Encodes 2nd order statistics

ICA:



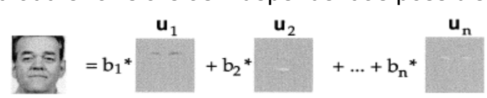
Independent TensorFaces:

Multilinear ICA

- Multilinear decomposition
- Encodes higher order statistics

ICA for face recognition: Architecture I statistical independent basis images

- Data matrix, D
  - images are observations and pixels are variables
- ICA finds a invertable transformation W such that the rows are as independent as possible



ICA representation = $(b_1, b_2, \dots, b_n)$

Fig. 7. The independent basis image representation consisted of the coefficients, b, for the linear combination of independent basis images, u, that comprised each face image x.

Architecture I: 25 IC

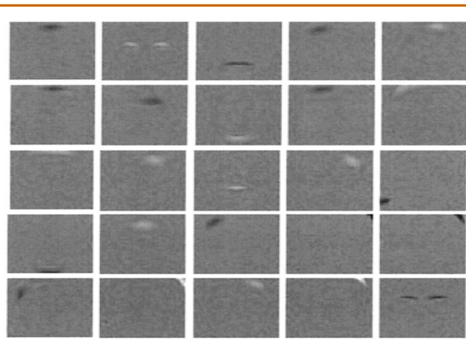
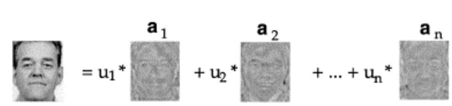


Fig. 8. Twenty-five ICs of the image set obtained by Architecture I, which provide a set of statistically independent basis images (rows of U in Fig. 4). ICs are ordered by the class discriminability ratio, r (4).

ICA face recognition: Architecture II Factorial Code

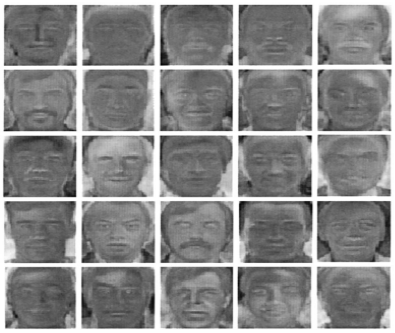
- Data matrix, **D**
  - pixels are observations
  - Images are variables



ICA factorial representation = $(u_1, u_2, \dots, u_n)$

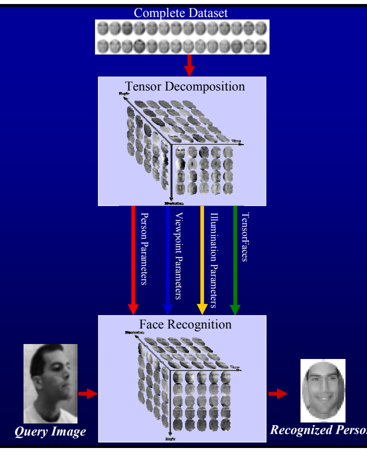
Fig. 12. The factorial code representation consisted of the independent coefficients, u, for the linear combination of basis images in A that comprised each face image x.

Architecture II: basis images

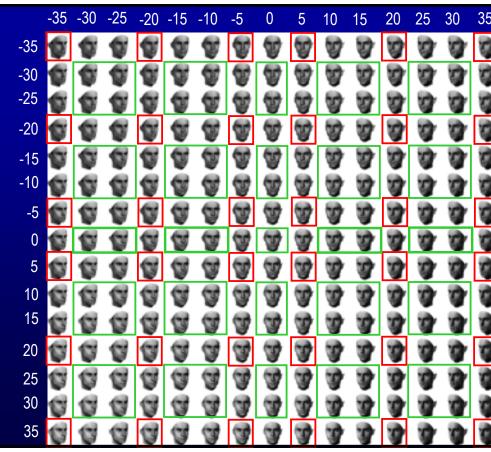


this approach tends to generate basis images that look more face-like than the basis images generated by PCA

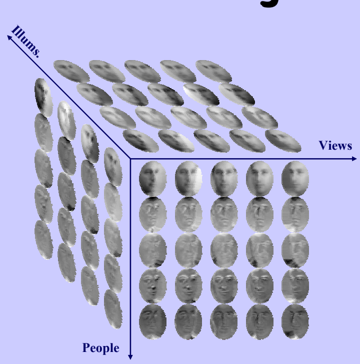
Fig. 13. Basis images for the ICA-factorial representation (columns of $A \triangleq W_I^{-1}$) obtained with Architecture II.



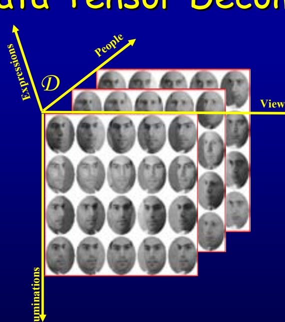
Freiburg 3D-Morphable Data



TensorFaces Recognition

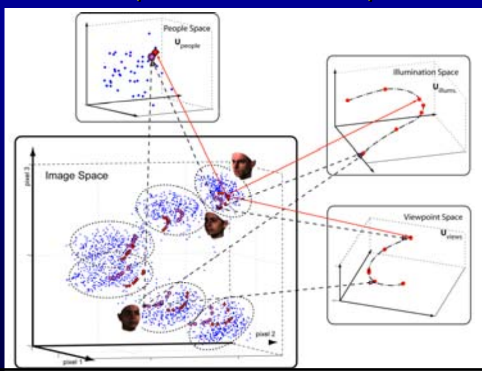


Facial Data Tensor Decomposition

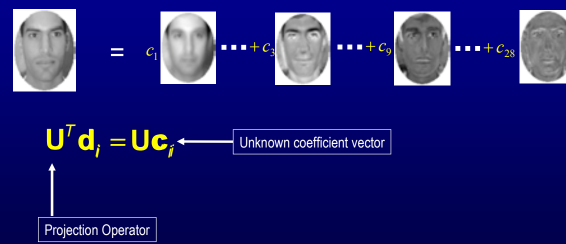


$\vec{d} = \mathcal{Z} \times_1 \mathbf{U}_{people} \times_2 \mathbf{U}_{views} \times_3 \mathbf{U}_{illums} \times_4 \mathbf{U}_{express} \times_5 \mathbf{U}_{pixels}$

Decomposition from Pixel Space to Factor Spaces



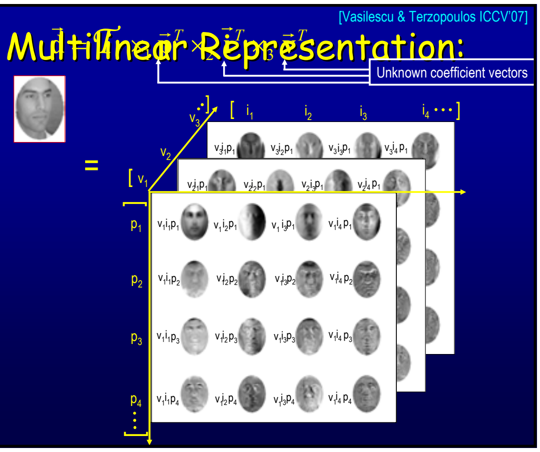
Linear Representation:



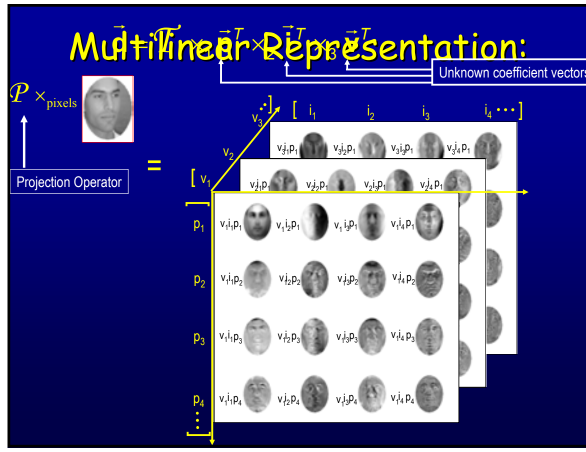
$$U^T d_i = U c_i$$

[Vasilescu & Terzopoulos ICCV'07]

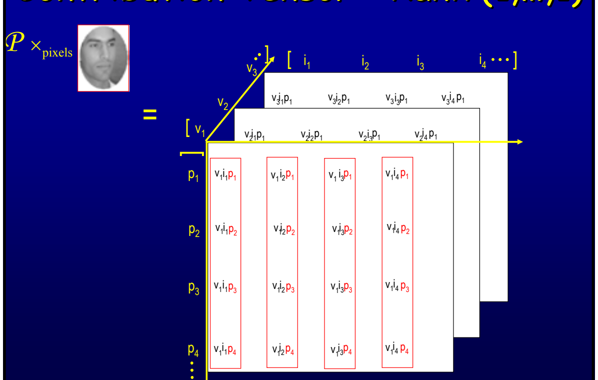
Multilinear Representation:



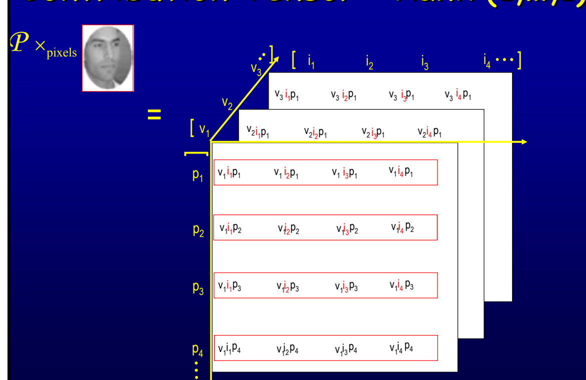
Multilinear Representation:



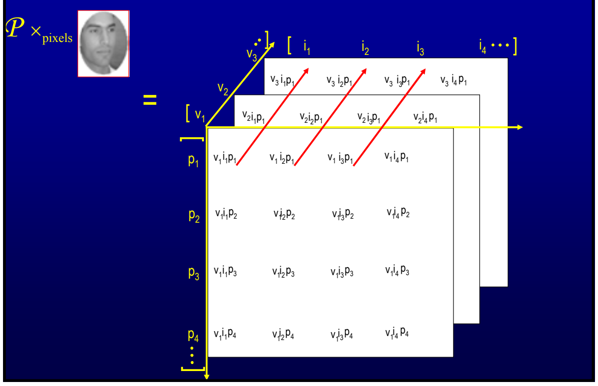
Contribution Tensor – Rank (1,…,1)



Contribution Tensor – Rank (1,…,1)



Contribution Tensor – Rank (1,…,1)



Contribution Tensor – Rank-(1,1,1)

Response Tensor

$$\underbrace{\mathcal{P} \times_{\text{pixels}} \mathbf{d}^T}_{} \approx \underbrace{\mathbf{p} \circ \mathbf{v} \circ \mathbf{l}}_{}$$

Projection Operator

where $\mathbf{P}_{(\text{pixels})} = \mathbf{T}^{+T}_{(\text{pixels})}$

Multilinear Projection

1. Compute the Projection Tensor:

$$\mathbf{P}_{(\text{observ. mode})} = \mathbf{T}^{+T}_{(\text{observ. mode})}$$

2. Compute the Response Tensor:

$$\mathcal{R} = \mathcal{P} \times_{\text{observ. mode}} \mathbf{d}^T$$
$$= \mathcal{I} \times_{\text{observ. mode}} (\mathbf{l}^T \otimes \mathbf{v}^T \otimes \mathbf{p}^T)$$
$$= (\mathbf{p} \circ \mathbf{v} \circ \mathbf{l})$$

3. Extract the coefficient vectors by factorizing the Response Tensor by using N-mode SVD.

Tensor-Matrix Relationships

- *1.* $\mathcal{B} = \mathcal{A} \times_n \mathbf{M} \quad \Longleftrightarrow \quad \mathbf{B}_{(n)} = \mathbf{M}\mathbf{A}_{(n)}$
- *2.* $\mathcal{D} = \mathcal{Z} \times_1 \mathbf{U}_1 \times_2 \mathbf{U}_2 \cdots \times_n \mathbf{U}_n \cdots \times_N \mathbf{U}_N$

$$\Updownarrow$$

$$\mathbf{D}_{(n)} = \mathbf{U}_n \mathbf{Z}_{(n)} (\mathbf{U}_{n-1} \otimes \cdots \otimes \mathbf{U}_1 \otimes \mathbf{U}_N \cdots \otimes \mathbf{U}_{n+2} \otimes \mathbf{U}_{n+1})^T$$

- *3. Kronecker product*

$$\mathbf{A} \otimes \mathbf{B} = \begin{bmatrix} a_{11} & a_{1m} \\ a_{n1} & a_{nm} \end{bmatrix} \otimes \mathbf{B} = \begin{bmatrix} a_{11}\mathbf{B} & a_{1m}\mathbf{B} \\ a_{n1}\mathbf{B} & a_{nm}\mathbf{B} \end{bmatrix}$$

Derivation

$$\mathcal{d} = \mathcal{T} \times_1 \mathbf{p}^T \times_2 \mathbf{v}^T \times_3 \mathbf{l}^T$$

$$\mathcal{d}_{(\text{pixels})} = (\mathcal{T} \times_1 \mathbf{p}^T \times_2 \mathbf{v}^T \times_3 \mathbf{l}^T)_{(\text{pixels})}$$

$$\mathbf{d} = \mathbf{T}_{(\text{pixels})} (\mathbf{l}^T \otimes \mathbf{v}^T \otimes \mathbf{p}^T)^T$$
$$\mathbf{d}^T = (\mathbf{l}^T \otimes \mathbf{v}^T \otimes \mathbf{p}^T) \mathbf{T}^T_{(\text{pixels})}$$
$$\mathbf{d}^T \mathbf{T}^{+T}_{(\text{pixels})} = (\mathbf{l}^T \otimes \mathbf{v}^T \otimes \mathbf{p}^T) \mathbf{T}^T_{(\text{pixels})} \mathbf{T}^{+T}_{(\text{pixels})}$$
$$\approx (\mathbf{l}^T \otimes \mathbf{v}^T \otimes \mathbf{p}^T) \mathbf{I}_{(\text{pixels})}$$

Tensor-Matrix Relationships

$$\mathcal{D} = \mathcal{Z} \times_1 \mathbf{U}_1 \times_2 \mathbf{U}_2 \cdots \times_n \mathbf{U}_n \cdots \times_N \mathbf{U}_N$$
$$\Updownarrow$$
$$\mathbf{D}_{(n)} = \mathbf{U}_n \mathbf{Z}_{(n)} (\mathbf{U}_{n-1} \otimes \cdots \mathbf{U}_1 \otimes \mathbf{U}_N \otimes \cdots \mathbf{U}_{n+1})^T$$

$$\underbrace{\mathcal{P} \times_{\text{pixels}} \mathbf{d}^T}_{\mathcal{R}} \approx \underbrace{\mathcal{I} \times_{\text{pixels}} (\mathbf{l}^T \otimes \mathbf{v}^T \otimes \mathbf{p}^T)}_{\substack{\mathcal{C} \\ (\mathbf{p} \circ \mathbf{v} \circ \mathbf{l})}}$$