The Secret of Rotating Object Images

Using Cyclic Permutation for View-based Pose Estimation

ACCV2007 Workshop on Subspace 2007
19, Nov. 2007 at the University of Tokyo, Tokyo Japan

Toru Tamaki, HIROSHIMA UNIVERSITY
Toshiyuki Amano, NAIST
Kazufumi Kaneda HIROSHIMA UNIVERSITY

The Key: Cyclic Permutation

The relationship between images

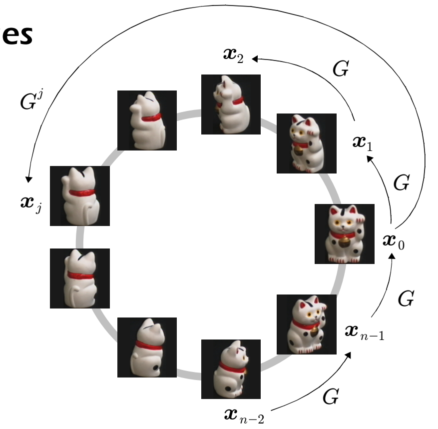
$$\boldsymbol{x}_{j+1 \bmod n} = G \boldsymbol{x}_j$$

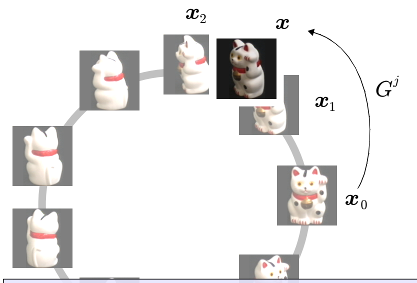
A matrix G transforms an image to another.

$$\boldsymbol{x}_1 = G \boldsymbol{x}_0$$
$$\boldsymbol{x}_2 = G \boldsymbol{x}_1$$
$$\vdots$$
$$\boldsymbol{x}_0 = G \boldsymbol{x}_{n-1}$$

G^j transforms x_0 to x_j directly

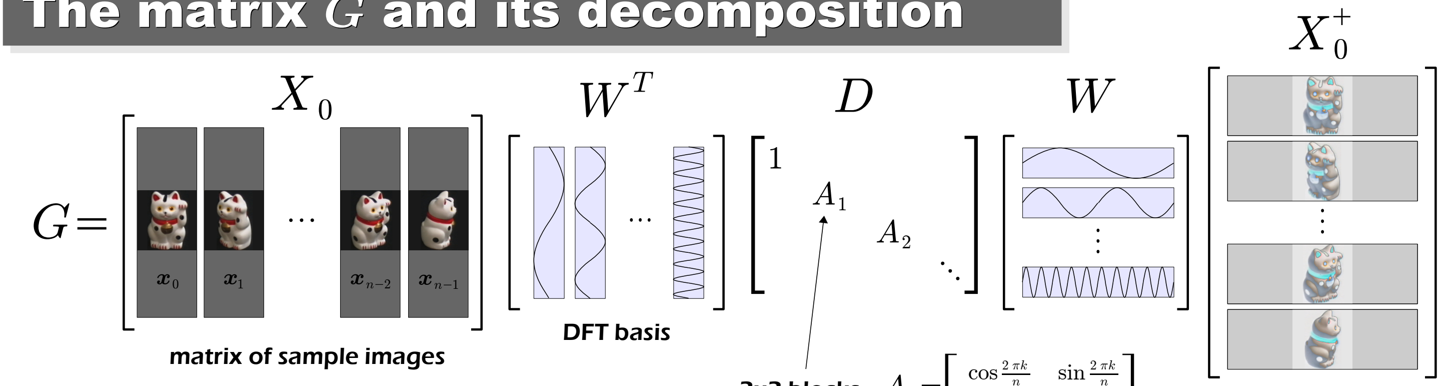
$$\boldsymbol{x}_j = G^j \boldsymbol{x}_0$$





A novel image x is represented as $\boldsymbol{x} = G^j \boldsymbol{x}_0$ for some j

The matrix G and its decomposition



matrix of sample images

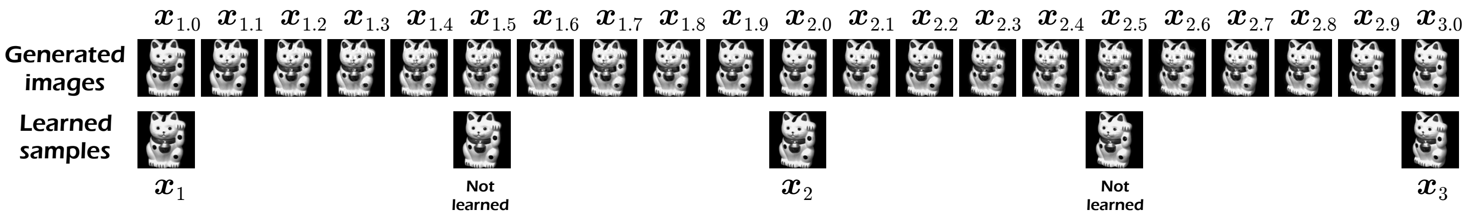
DFT basis

2x2 blocks $A_k = \begin{bmatrix} \cos\frac{2\pi k}{n} & \sin\frac{2\pi k}{n} \\ -\sin\frac{2\pi k}{n} & \cos\frac{2\pi k}{n} \end{bmatrix}$

Pseudo inverse

View Generation

Learn finite samples, Generate many images : $\boldsymbol{x}_{1+0.1j} = G^{0.1j} \boldsymbol{x}_1 = X_0 W^T D^{0.1j} W X_0^+ \boldsymbol{x}_1$



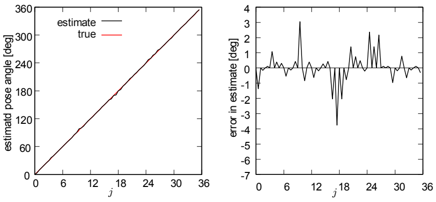
Pose Estimation

Compare images projected onto the subspace :

$$\boldsymbol{x} = G^j \boldsymbol{x}_0$$
$$\boldsymbol{x} = X_0 W^T D^j W X_0^+ \boldsymbol{x}_0$$
$$W X_0^+ \boldsymbol{x} = D^j W X_0^+ \boldsymbol{x}_0$$
$$\boldsymbol{x}' = D^j \boldsymbol{x}_0'$$

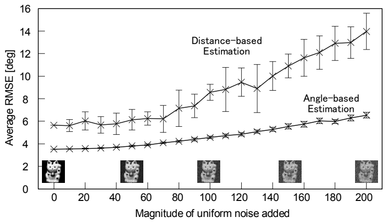
Find j such that two images in the subspace are close to each other

image in the subspace
$\boldsymbol{x}' = W X_0^+ \boldsymbol{x}$

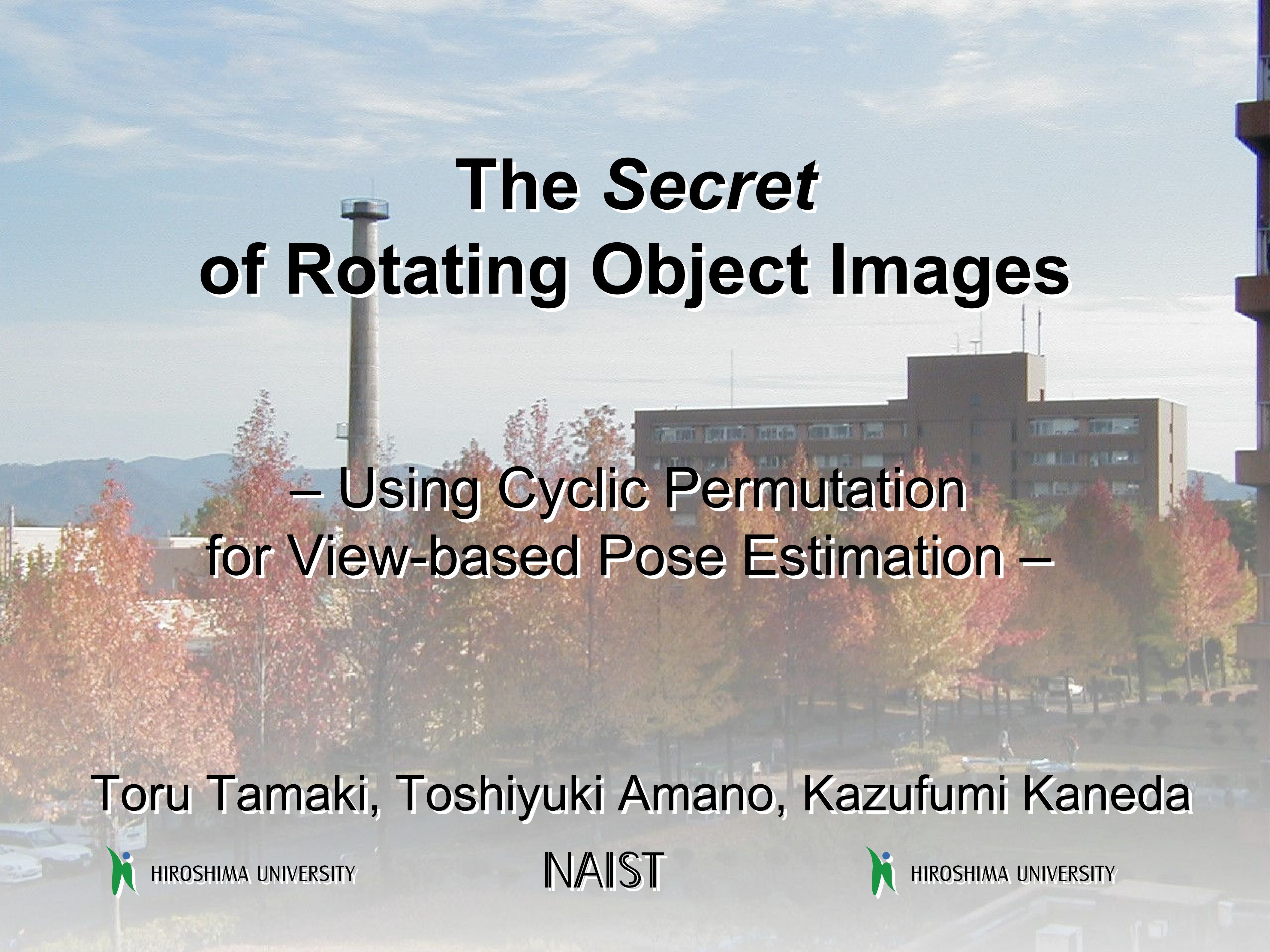


Target :
obj4 in COIL-20

Error :
Max 3.76[deg]
RMSE 1.23[deg]



Estimation error under uniform noise for 20 objects in COIL-20



The *Secret* of Rotating Object Images

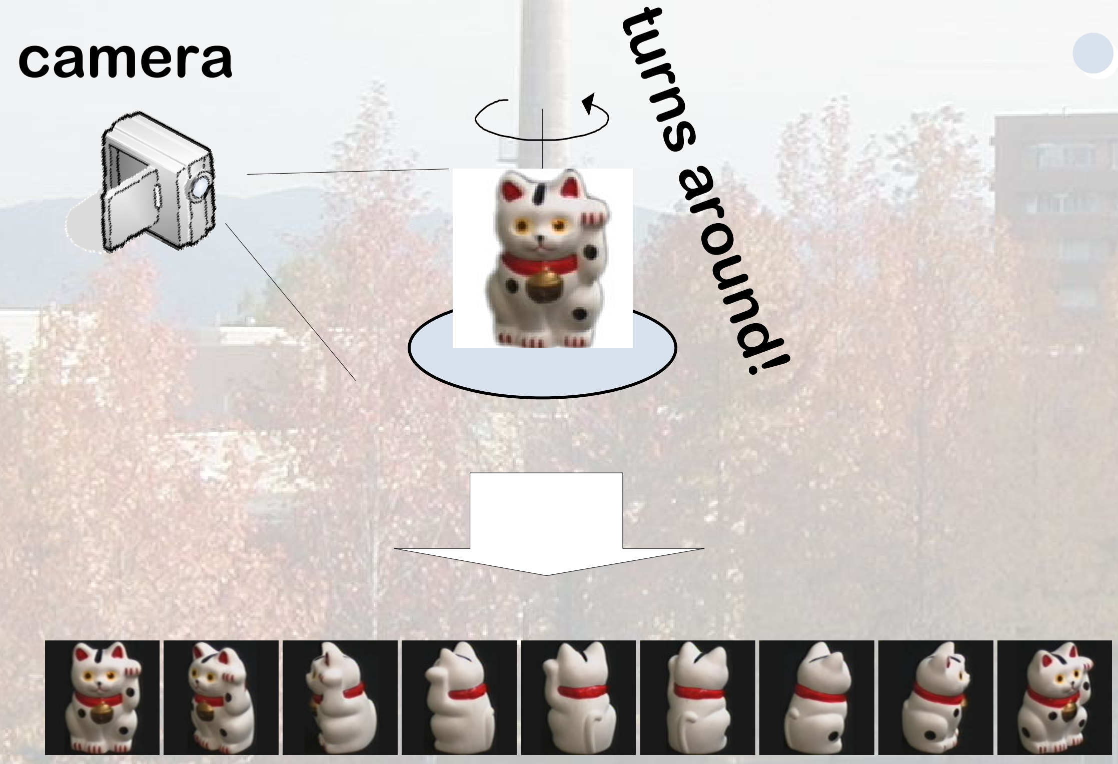
– Using Cyclic Permutation for View-based Pose Estimation –

Toru Tamaki, Toshiyuki Amano, Kazufumi Kaneda

HIROSHIMA UNIVERSITY NAIST HIROSHIMA UNIVERSITY

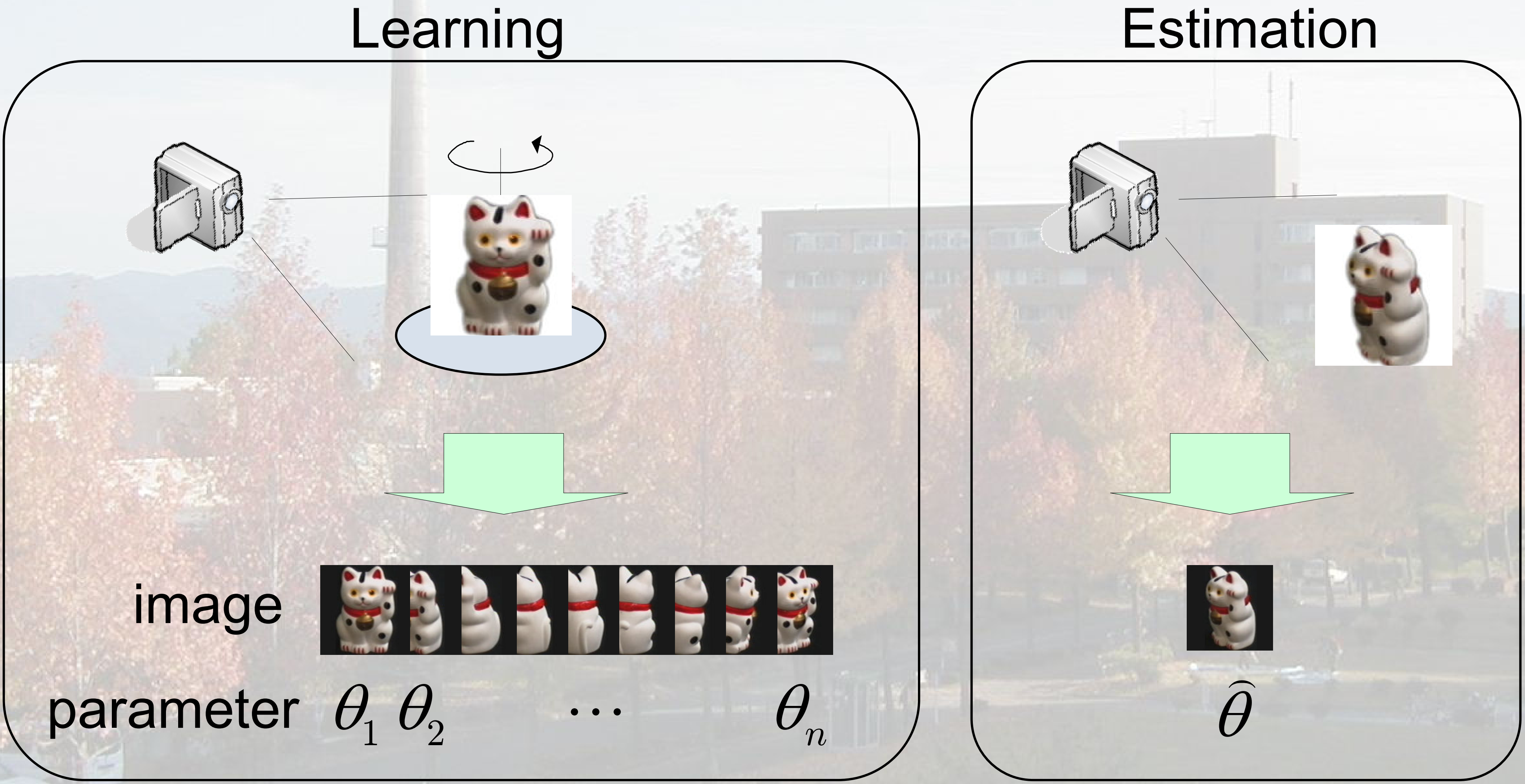
When an object rotates…

- Images are taken.
- They make a manifold…?



What are these images?

View-based pose estimation



Learning

image

parameter θ_1 θ_2 $\cdots$ θ_n

Estimation

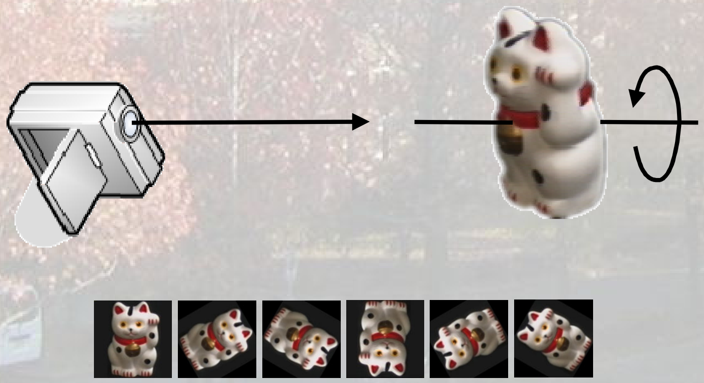
$\hat{\theta}$

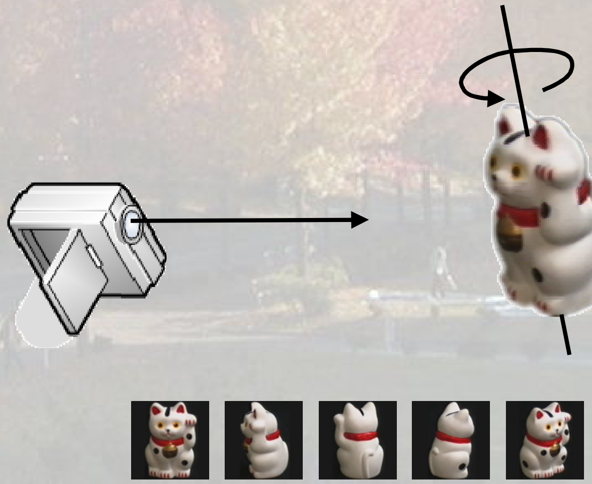
- Learns the relationship between images (input) and parameters (output)

In and Off: 1DOF rotations

- in-plane rotation
  - about the optical axis of the camera
  - appearance rotates in the image plane

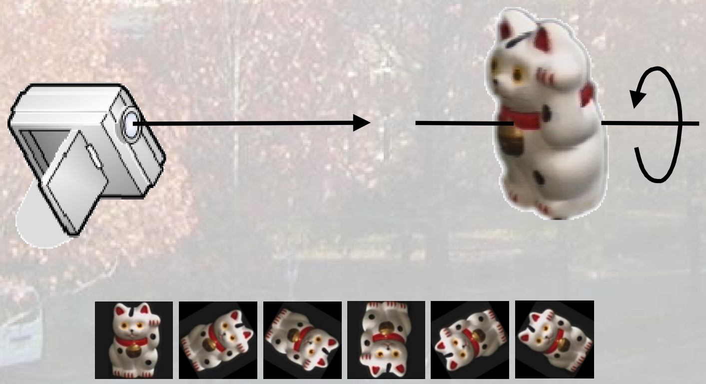
- off-the-plane rotation
  - about any axis in 3D
  - appearance changes due to the object geometry





In and Off: 1DOF rotations

- **in-plane rotation**
  - about the optical axis of the camera
  - appearance rotates in the image plane

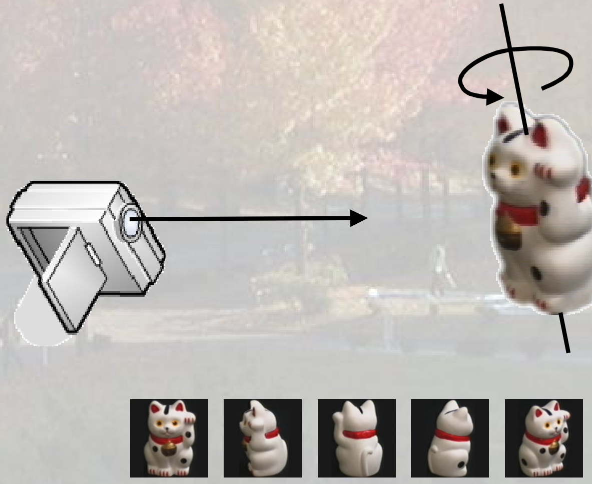


- Analitical solutions
  - Eigenimages are obtained by DFT.
    - (Uenohara et al., 1998)
    - (Chang et al., 2000)
    - (Park, 2002)
    - (Jorgan et al., 2003)
    - (Sengel et al., 2005)
  - But, hopeless for extending off-the-plane rotation.

In and Off: 1DOF rotations

- Pose estimation
  - Parametric Eigenspace method
    - (Murase et al., 1995)
  - linear regression
    - (Okatani et al., 2000)
    - (Amano et al., 2007)
  - kernel methods
    - (Melzer et al., 2003)
    - (Ando et al., 2005)
  - Manifold learning

- **off-the-plane rotation**
  - about any axis in 3D
  - appearance changes due to the object geometry

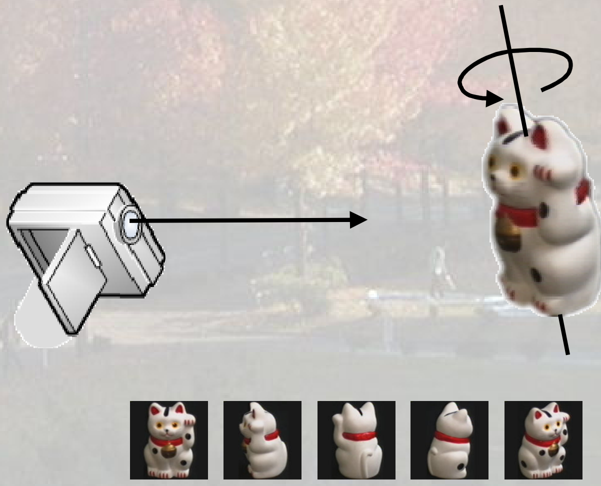


In and Off: 1DOF rotations

- Questions:
  - Can we represent these images analytically?
  - and How?

- What is *the key* to understand?
  - in terms of “linear”

- **off-the-plane rotation**
  - about any axis in 3D
  - appearance changes due to the object geometry



The Key : Cyclic Permutation.

The relationship between images:

$$\boldsymbol{x}_{j+1 \bmod n} = G\boldsymbol{x}_j$$

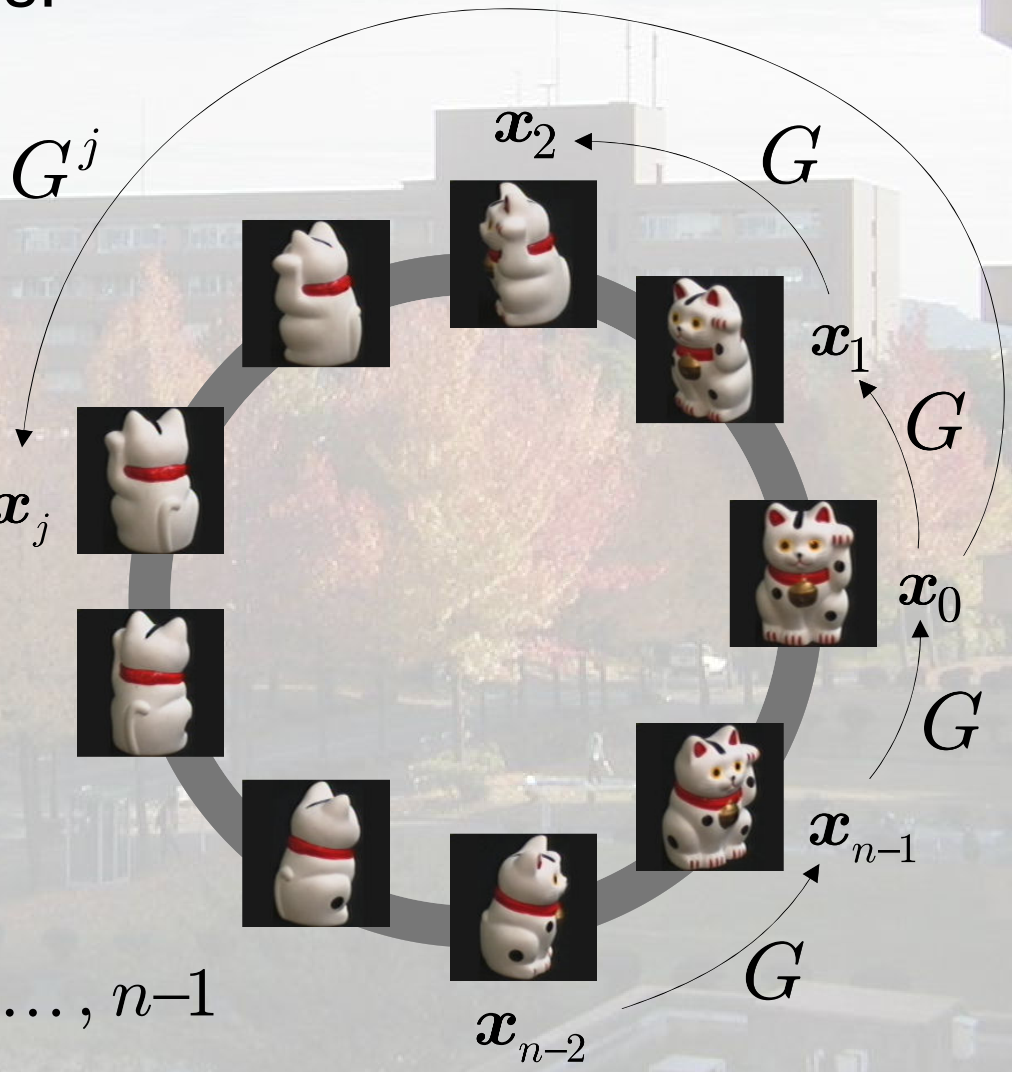
A matrix G transforms an image to another:

$$\begin{aligned} \boldsymbol{x}_1 &= G\boldsymbol{x}_0 \\ \boldsymbol{x}_2 &= G\boldsymbol{x}_1 \\ &\vdots \\ \boldsymbol{x}_0 &= G\boldsymbol{x}_{n-1} \end{aligned}$$

G^j transforms $\boldsymbol{x}_0$ to $\boldsymbol{x}_j$ directly.

$$\boldsymbol{x}_j = G^j\boldsymbol{x}_0$$

$j = 0, 1, 2, \ldots, n-1$



Why Cyclic Permutation?

- Pose Estimation
  - Find j such that

$$\boldsymbol{x} = G^j \boldsymbol{x}_0$$

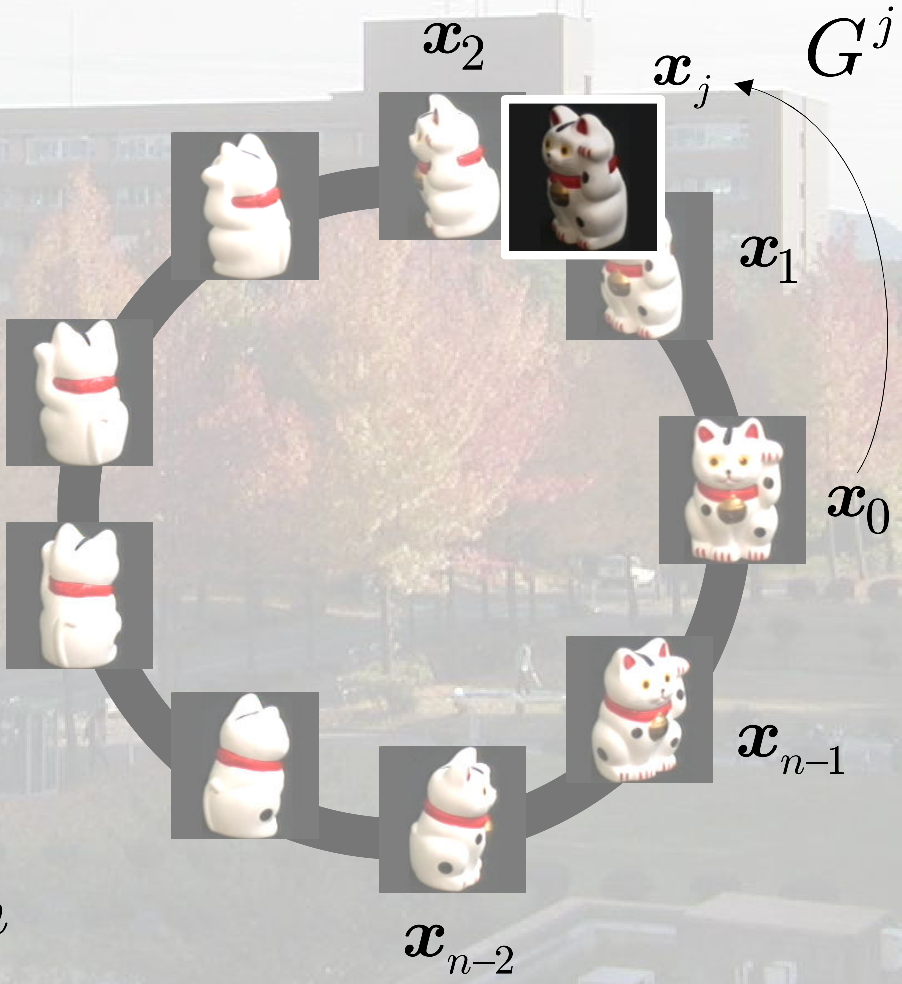
for a given image $\boldsymbol{x}$

- View Generation
  - Create an image $\boldsymbol{x}_j$

$$\boldsymbol{x}_j = G^j \boldsymbol{x}_0$$

for given j

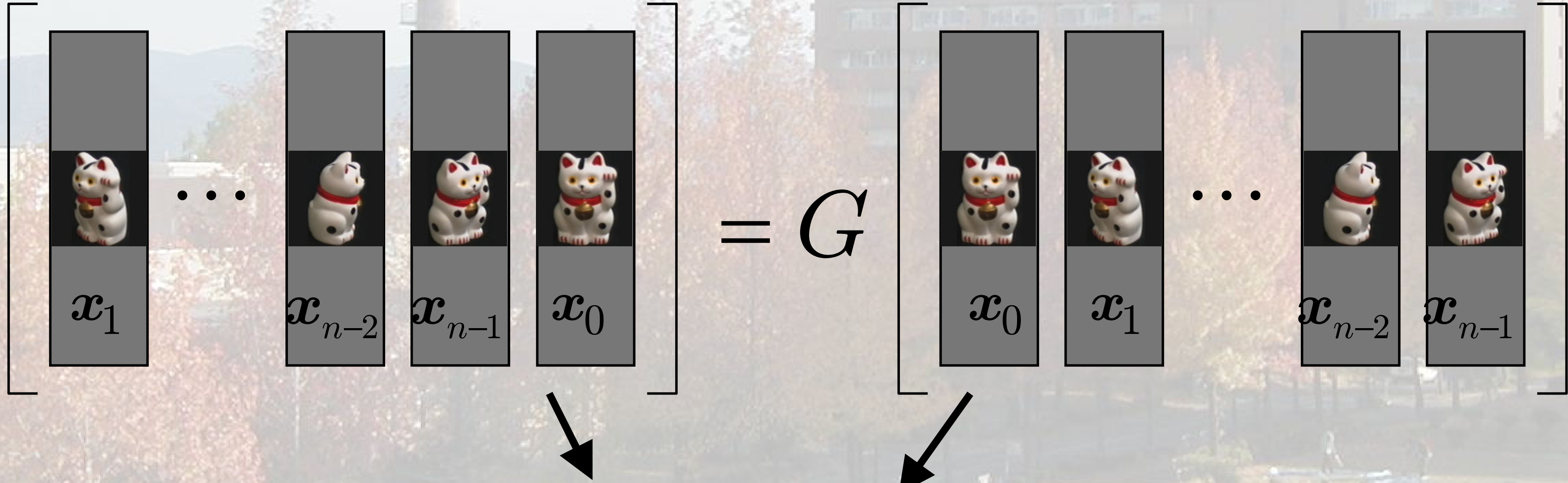
$0 \leq j < n$



Obtaining the matrix G …

Matrix representation

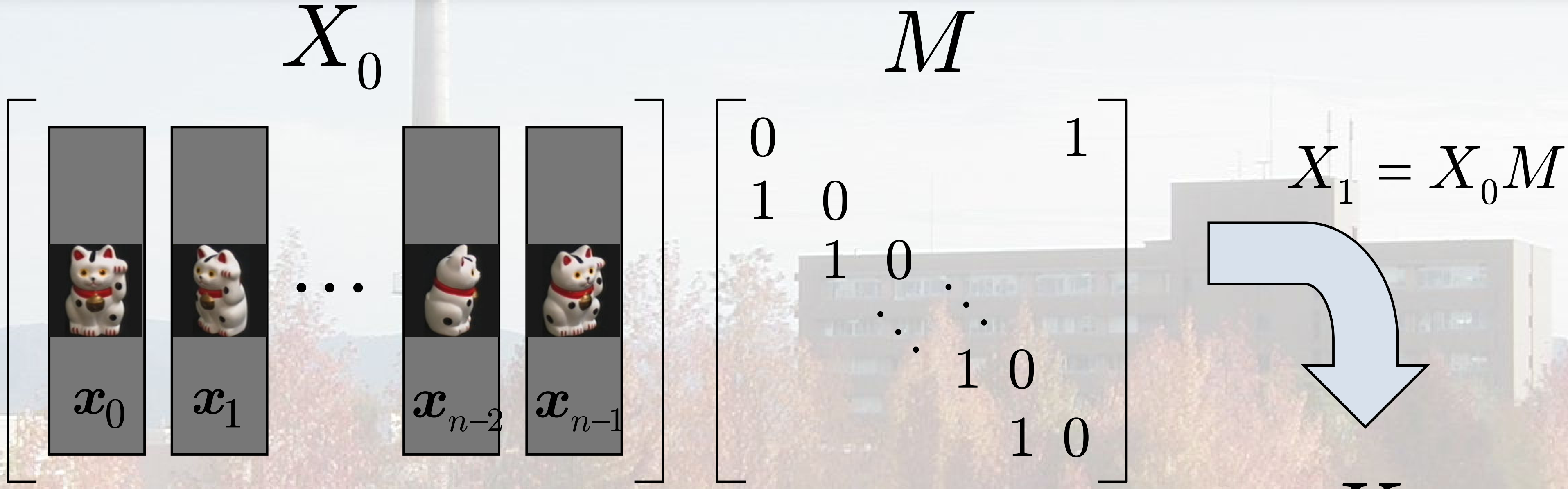
$$\left[\boldsymbol{x}_1\ \boldsymbol{x}_2 \ldots \boldsymbol{x}_{n-1}\ \boldsymbol{x}_0\right] = G\left[\boldsymbol{x}_0\ \boldsymbol{x}_1 \ldots \boldsymbol{x}_{n-2}\ \boldsymbol{x}_{n-1}\right]$$



$$X_1 = GX_0 \Rightarrow G = X_1X_0^+$$

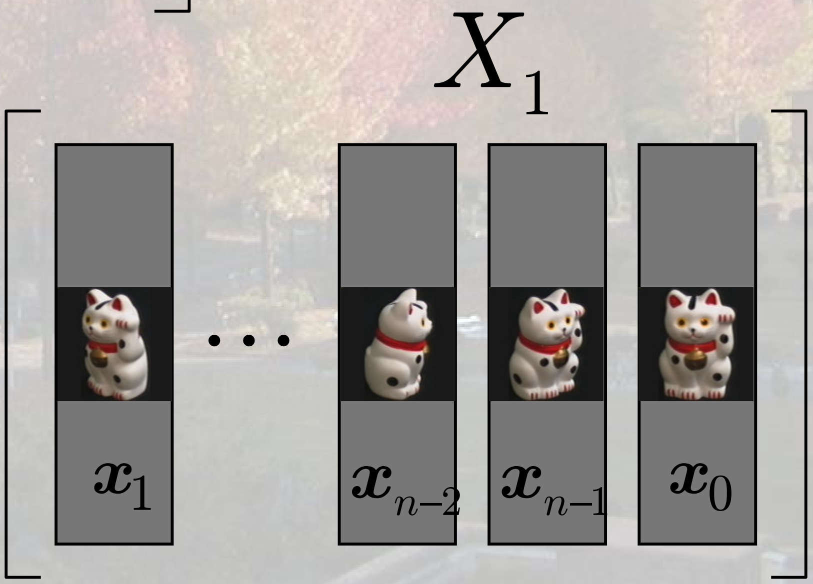
Using pseudoinverse

with Column permutation.



$$M = \begin{bmatrix} 0 & & & & & 1 \\ 1 & 0 & & & & \\ & 1 & 0 & \ddots & & \\ & & \ddots & \ddots & & \\ & & & 1 & 0 & \\ & & & & 1 & 0 \end{bmatrix}$$

$$X_1 = X_0 M$$



$$G = X_1 X_0^+ = X_0 M X_0^+$$

But,

Does really G transform $\boldsymbol{x}_0$ to $\boldsymbol{x}_1$?

$$\boxed{G \ = \ X_0 M X_0^+}$$

$$\begin{tikzcd}
X_1 & X_0 \arrow[l, "G"']
\end{tikzcd}$$

Yes, it does !

$$\begin{tikzcd}
X_1 & M \arrow[l, "X_0"'] & I \arrow[l, "M"'] & X_0 \arrow[l, "X_0^+"']
\end{tikzcd}$$

$$\begin{bmatrix} 0 & & & & & 1 \\ 1 & 0 & & & & \\ & 1 & 0 & \ddots & & \\ & & \ddots & \ddots & & \\ & & & 1 & 0 & \\ & & & & 1 & 0 \end{bmatrix} \qquad \begin{bmatrix} 1 & & & & & \\ & 1 & & & & \\ & & 1 & & & \\ & & & \ddots & & \\ & & & & 1 & \\ & & & & & 1 \end{bmatrix}$$

Block diagonalization

$$G = X_0 M X_0^+ = X_0 \begin{bmatrix} 0 & & & & & 1 \\ 1 & 0 & & & & \\ & 1 & 0 & \ddots & & \\ & & \ddots & \ddots & & \\ & & & 1 & 0 & \\ & & & & 1 & 0 \end{bmatrix} X_0^+$$

$$= X_0 \, W^T \underbrace{\begin{bmatrix} 1 & & & \\ & \boxed{A_1} & & \\ & & \boxed{A_2} & \\ & & & \ddots \end{bmatrix}}_{D} W \, X_0^+$$

DFT basis

2x2 blocks
(rotation matrix)

Decomposing the matrix G

$$G = X_0 W^T \underset{D}{\begin{bmatrix} 1 & & & \\ & A_1 & & \\ & & A_2 & \\ & & & \ddots \end{bmatrix}} W X_0^+$$

$$\Downarrow \qquad\qquad\qquad\qquad \Downarrow$$

$$G = U_2 \quad D \quad U_1$$

$$\boxed{G^j = U_2 \quad D^j \quad U_1}$$

View generation

Equations for leaning samples

$$\boldsymbol{x}_j = G^j \boldsymbol{x}_0 = U_2 D^j U_1 \boldsymbol{x}_0 \qquad j = 0, 1, 2, \ldots, n–1$$

Extend j to arbitrary number

$$\boldsymbol{x}_j = G^j \boldsymbol{x}_0 = U_2 D^j U_1 \boldsymbol{x}_0 \qquad 0 \leq j < n$$

Using only finite number of images $\boldsymbol{x}_0$, $\boldsymbol{x}_1$, …, $\boldsymbol{x}_{n\text{-}1}$,

Generating $\boldsymbol{x}_j$ for any j .

Pose estimation

Equations for leaning samples

$$\boldsymbol{x}_j = G^j \boldsymbol{x}_0 = U_2 D^j U_1 \boldsymbol{x}_0 \qquad j = 0, 1, 2, \ldots, n–1$$

Estimating j of arbitrary number

$$\boldsymbol{x} = G^j \boldsymbol{x}_0 = U_2 D^j U_1 \boldsymbol{x}_0 \qquad 0 \leq j < n$$

$$U_1 \boldsymbol{x} = U_1 U_2 D^j U_1 \boldsymbol{x}_0$$

$$U_1 \boldsymbol{x} = D^j U_1 \boldsymbol{x}_0$$

$$\boldsymbol{x}' = D^j \boldsymbol{x}'_0$$

Compare two images
projected in the subspace

$$\boldsymbol{x}'_j = U_1 \boldsymbol{x}_j$$

An image
in the subspace

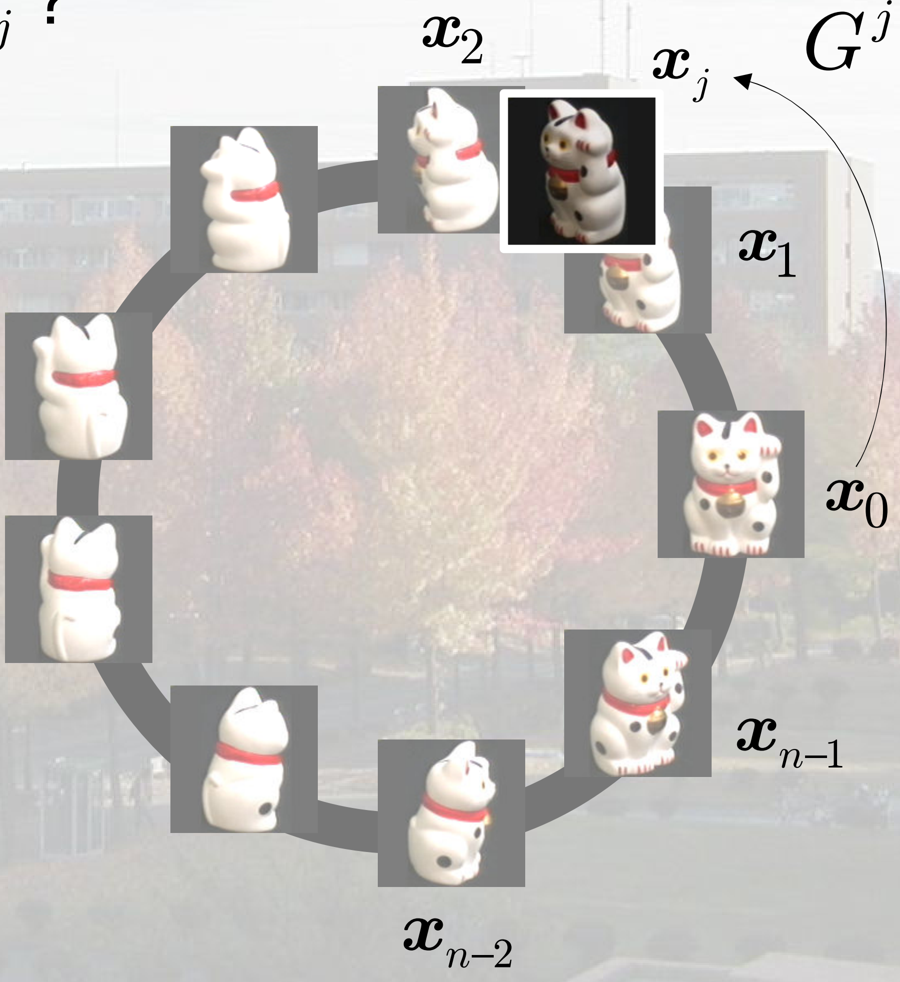
What’s the power of G?

Does G^j really transform $\boldsymbol{x}_0$ to $\boldsymbol{x}_j$?

$$G^j = U_2 D^j U_1$$

Can G^j really produce the back from the front?

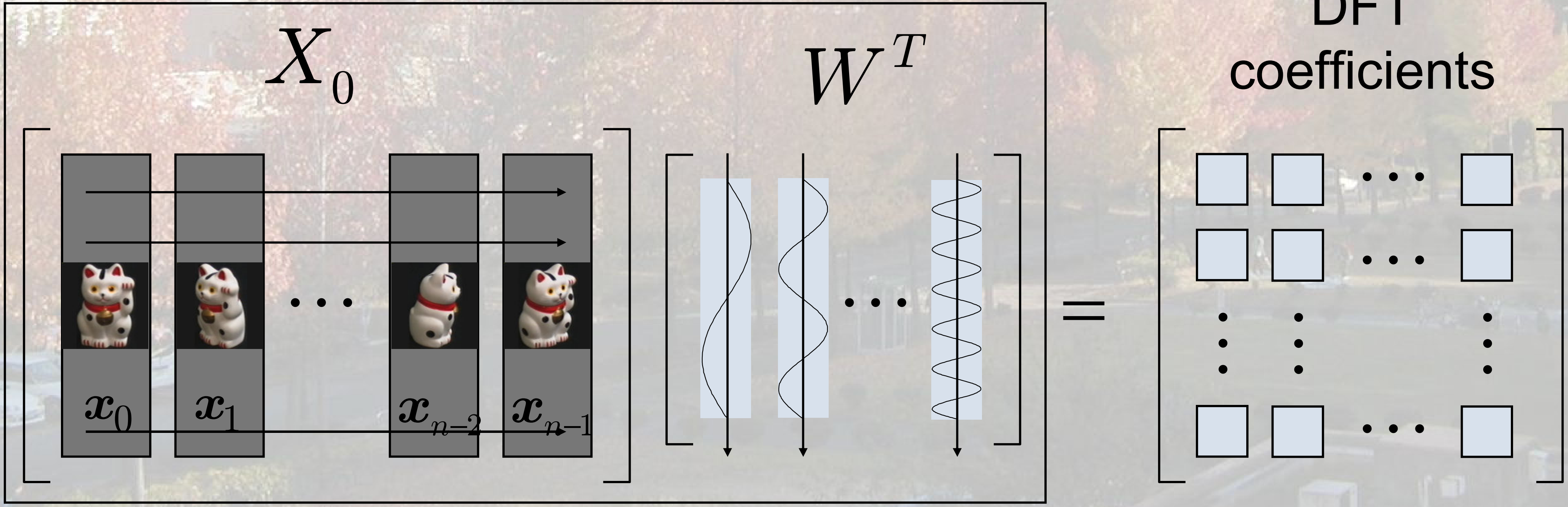
$$= G^j$$



Pixel-wise DFT

$$G = \boxed{X_0 W^T} \overset{D}{\begin{bmatrix} 1 & & & \\ & A_1 & & \\ & & A_2 & \\ & & & \ddots \end{bmatrix}} W X_0^+$$

DFT basis



DFT coefficients

Reconstruction by DFT basis

$j = 0$

$$G^0 = X_0 W^T \overset{D^0}{\begin{bmatrix} 1 & & & \\ & A_1^0 & & \\ & & A_2^0 & \\ & & & \ddots \end{bmatrix}} W X_0^+$$

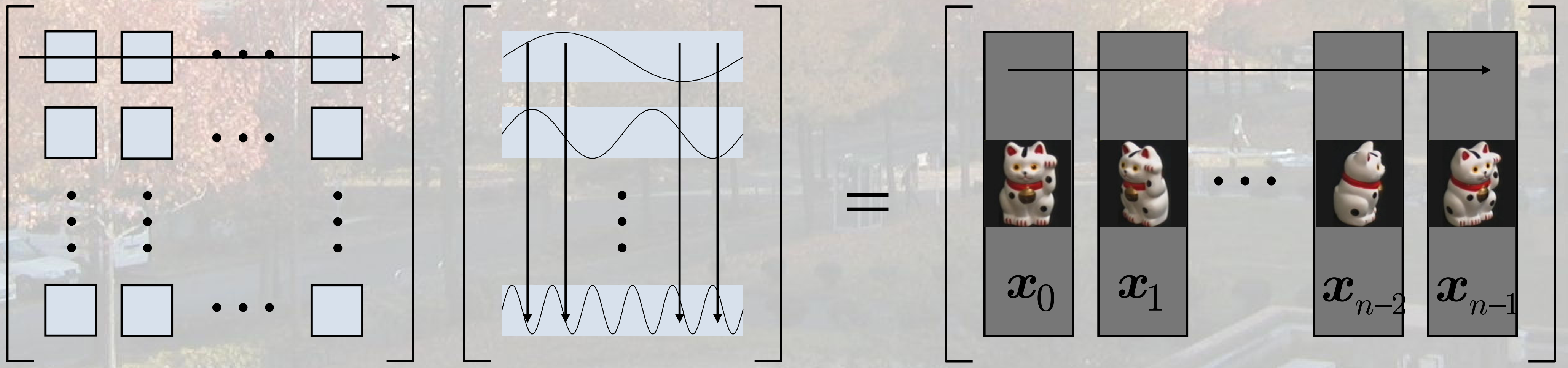
DFT basis

DFT coefficients

W

$D^0 = I$

X_0



Reconstruction by DFT basis

$j = 0$

$$G^0 = X_0 W^T \overset{D^0}{\begin{bmatrix} 1 & & & \\ & A_1^0 & & \\ & & A_2^0 & \\ & & & \ddots \end{bmatrix}} W X_0^+$$

DFT basis

X_0

$$\begin{bmatrix} \boxed{1} \\ \boxed{0} \\ \vdots \\ \boxed{0} \end{bmatrix} = X_0^+ \boldsymbol{x}_0$$

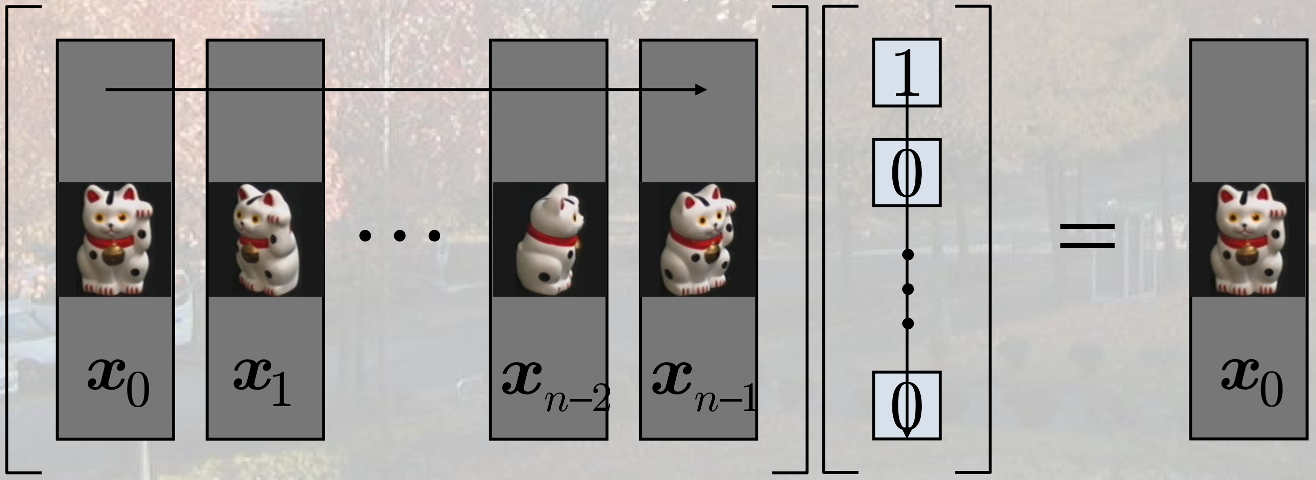
Reconstruction by DFT basis

$j = 0$

DFT basis

$$G^0 = X_0 W^T \overbrace{\begin{bmatrix} 1 & & & \\ & A_1^0 & & \\ & & A_2^0 & \\ & & & \ddots \end{bmatrix}}^{D^0} W X_0^+$$

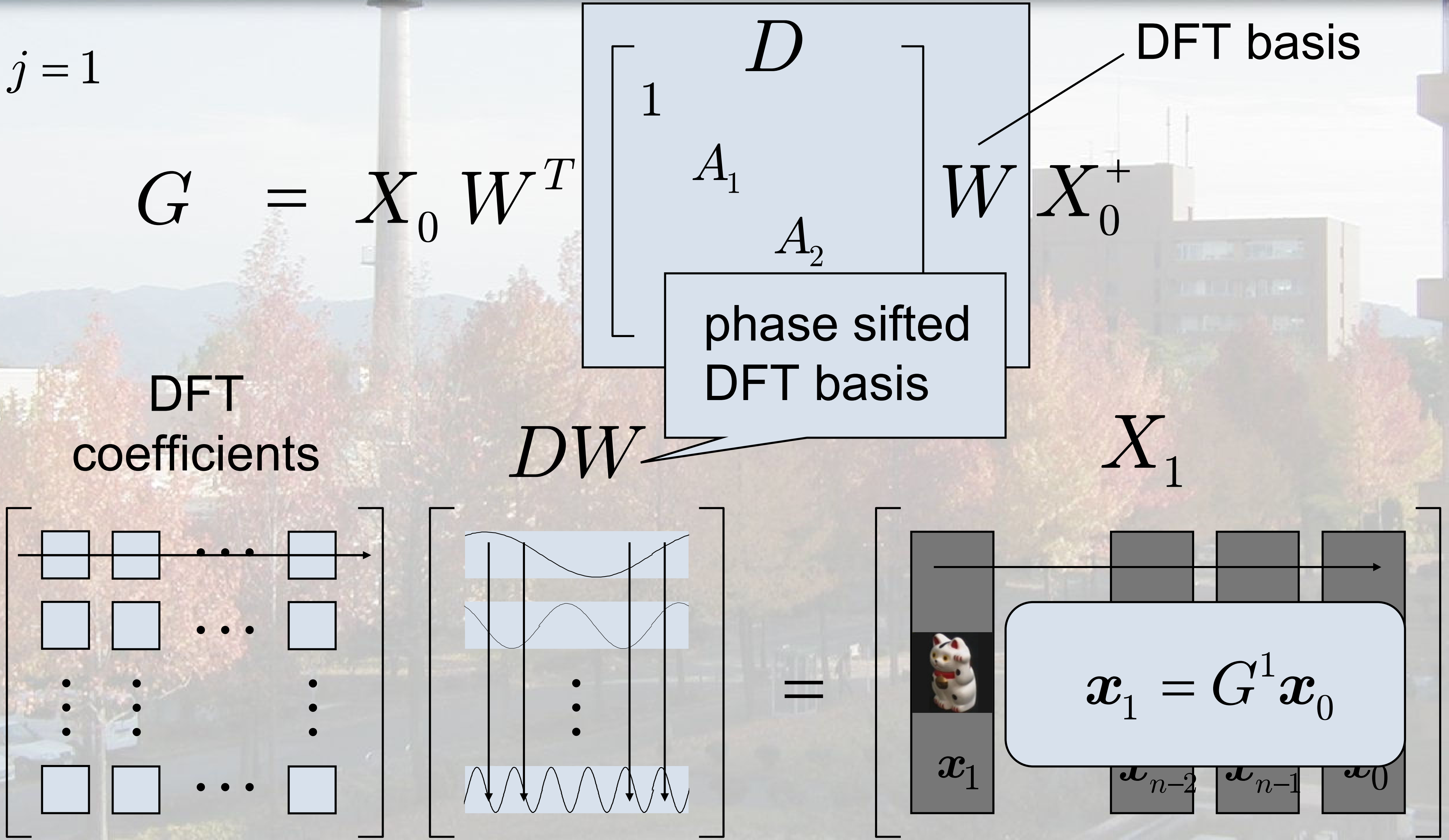
X_0



$$\boldsymbol{x}_0 = G^0 \boldsymbol{x}_0$$

Phase shift of DFT basis

$j = 1$



$$G = X_0 W^T \overset{D}{\begin{bmatrix}1&&\\&A_1&\\&&A_2\end{bmatrix}} W X_0^+$$

DFT basis

phase sifted DFT basis

DFT coefficients

$$\text{DFT coefficients} \; DW = X_1$$

$$\boldsymbol{x}_1 = G^1 \boldsymbol{x}_0$$

Continuous phase shift

$0 \le j < n$

$$G^j = X_0 W^T \overbrace{\begin{bmatrix} 1 & & \\ & A_1^j & \\ & & A_2^j \end{bmatrix}}^{D^j} W X_0^+$$

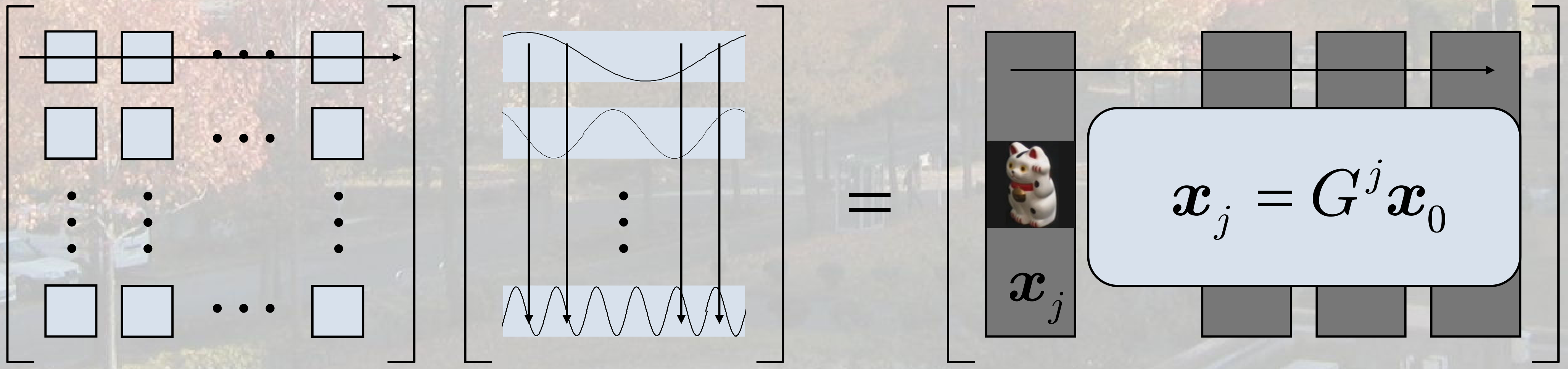
DFT bases

phase sifted DFT bases

DFT coefficients

$D^j W$

X_j



$$x_j = G^j x_0$$

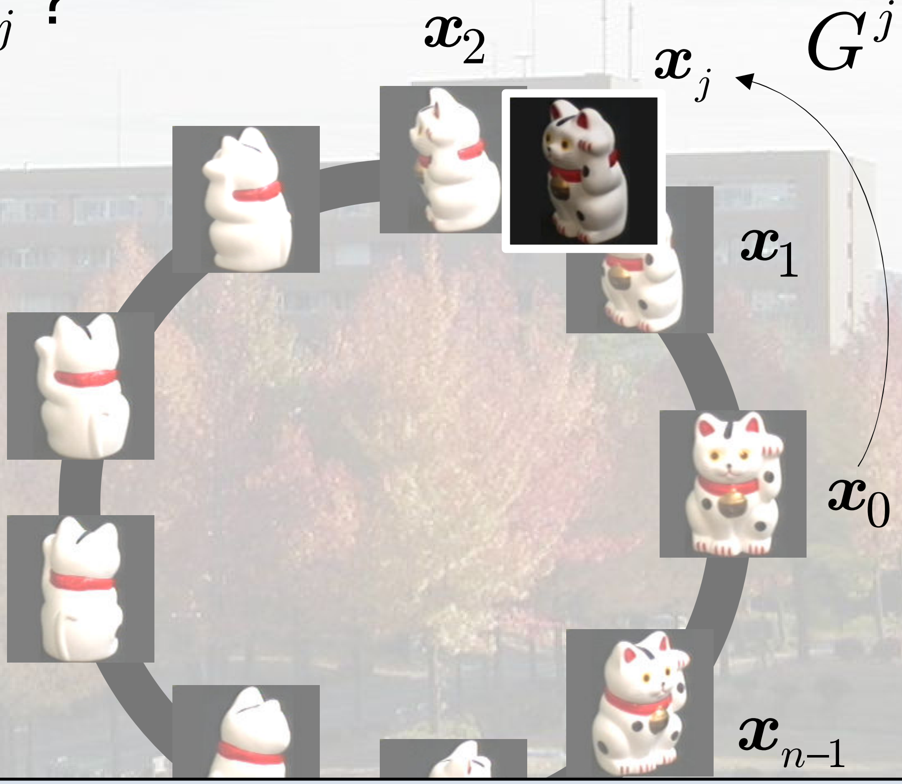
This is the power of G!

Does G^j really transform $\boldsymbol{x}_0$ to $\boldsymbol{x}_j$?

$$G^j = U_2 D^j U_1$$

Can G^j really produce
the back from the front?

$= G^j$



Yes, it does by pixel-wise DFT!

Conclusions

- Introduced cyclic permutation to represent images of rotationg object.
- Applied to view generation and pose estimation.
- Closely related to DFT in pixel-wise for generating novel image by $x_j = G^j x_0$.