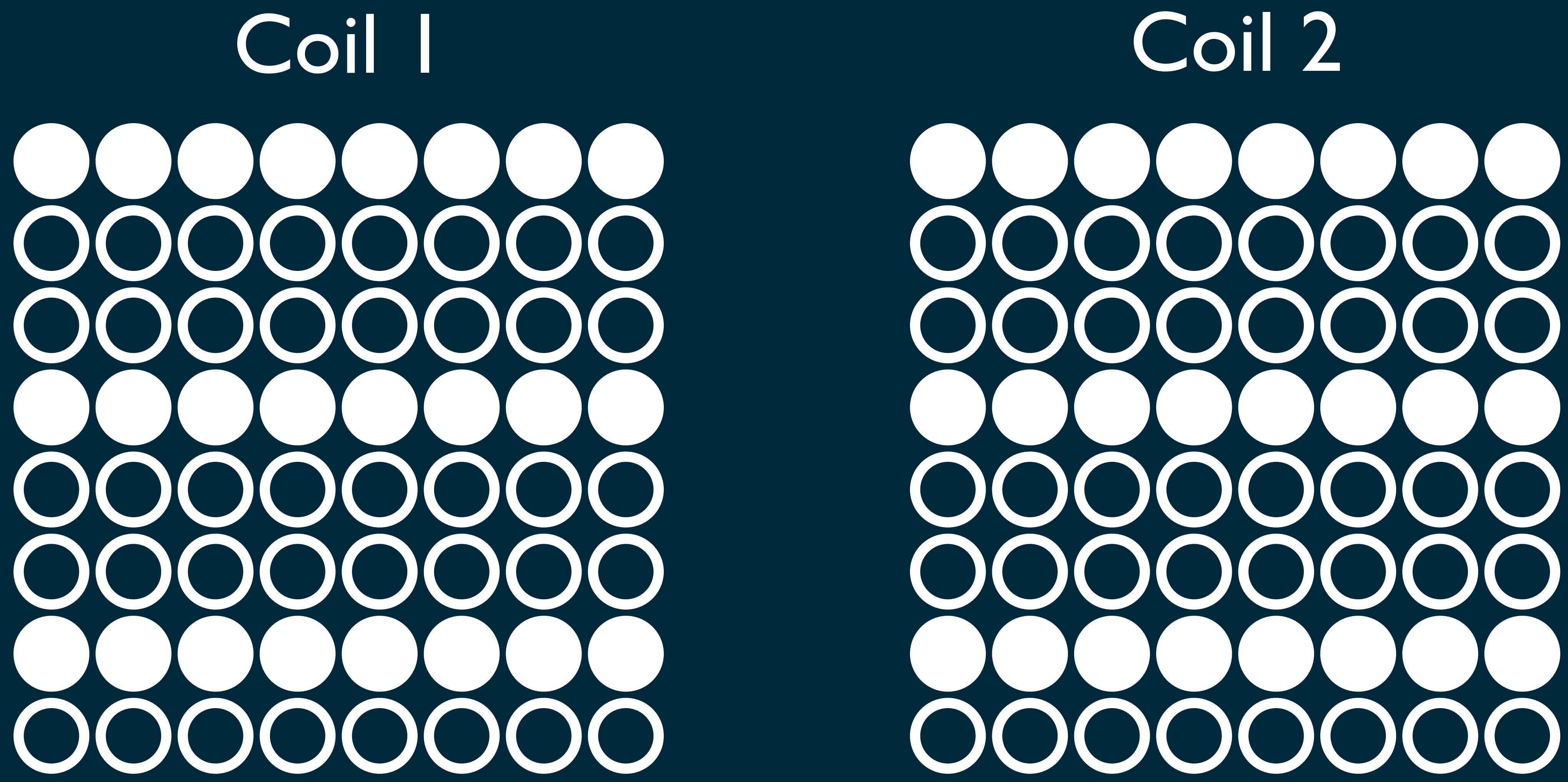
Learn how to GRAPPA

FMRIB Graduate Course
Advanced Program

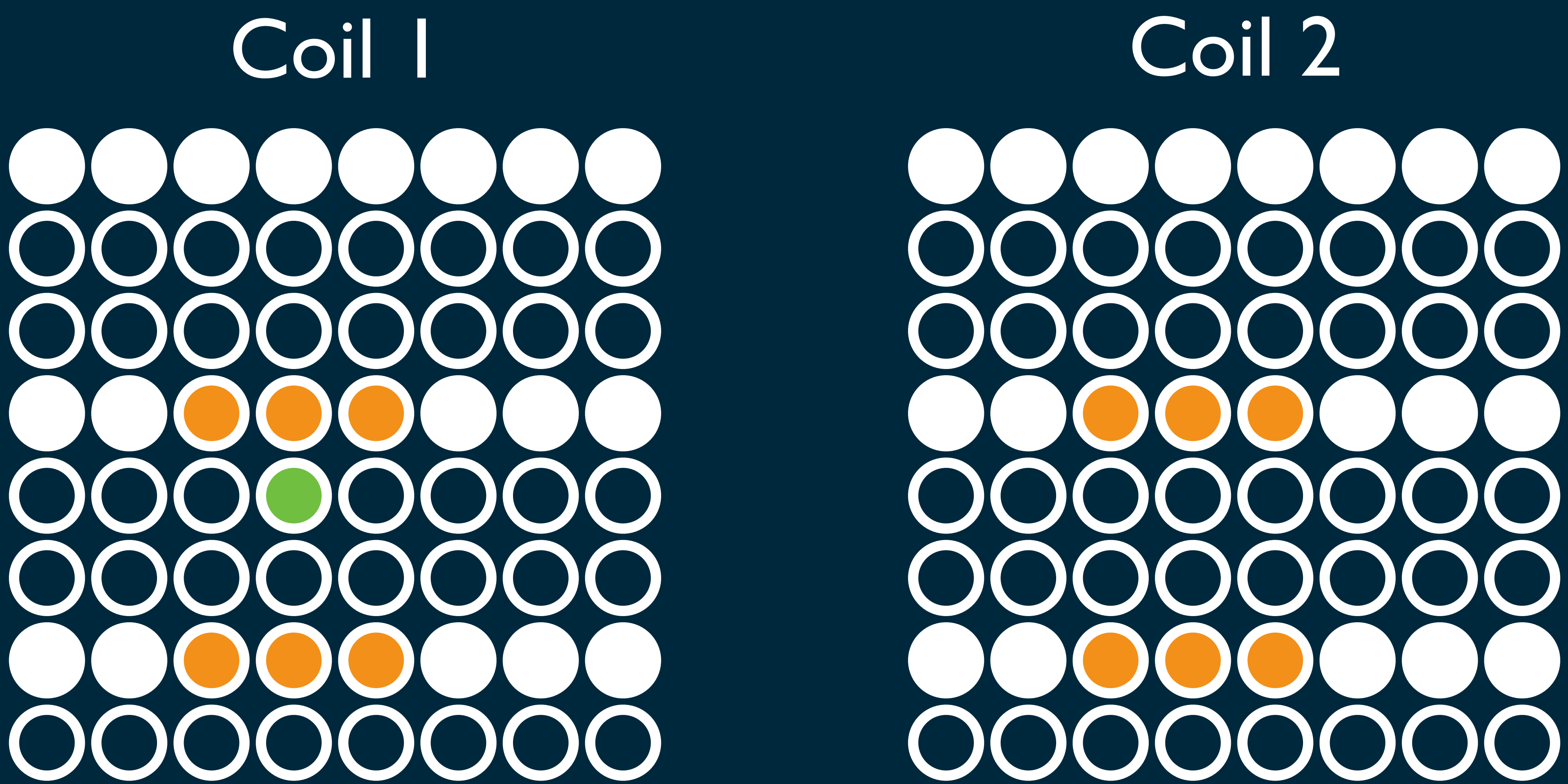
Mark Chiew (mchiew@fmrib.ox.ac.uk)
Feb 09, 2017

GRAPPA Geometry



Estimate missing points from acquired points

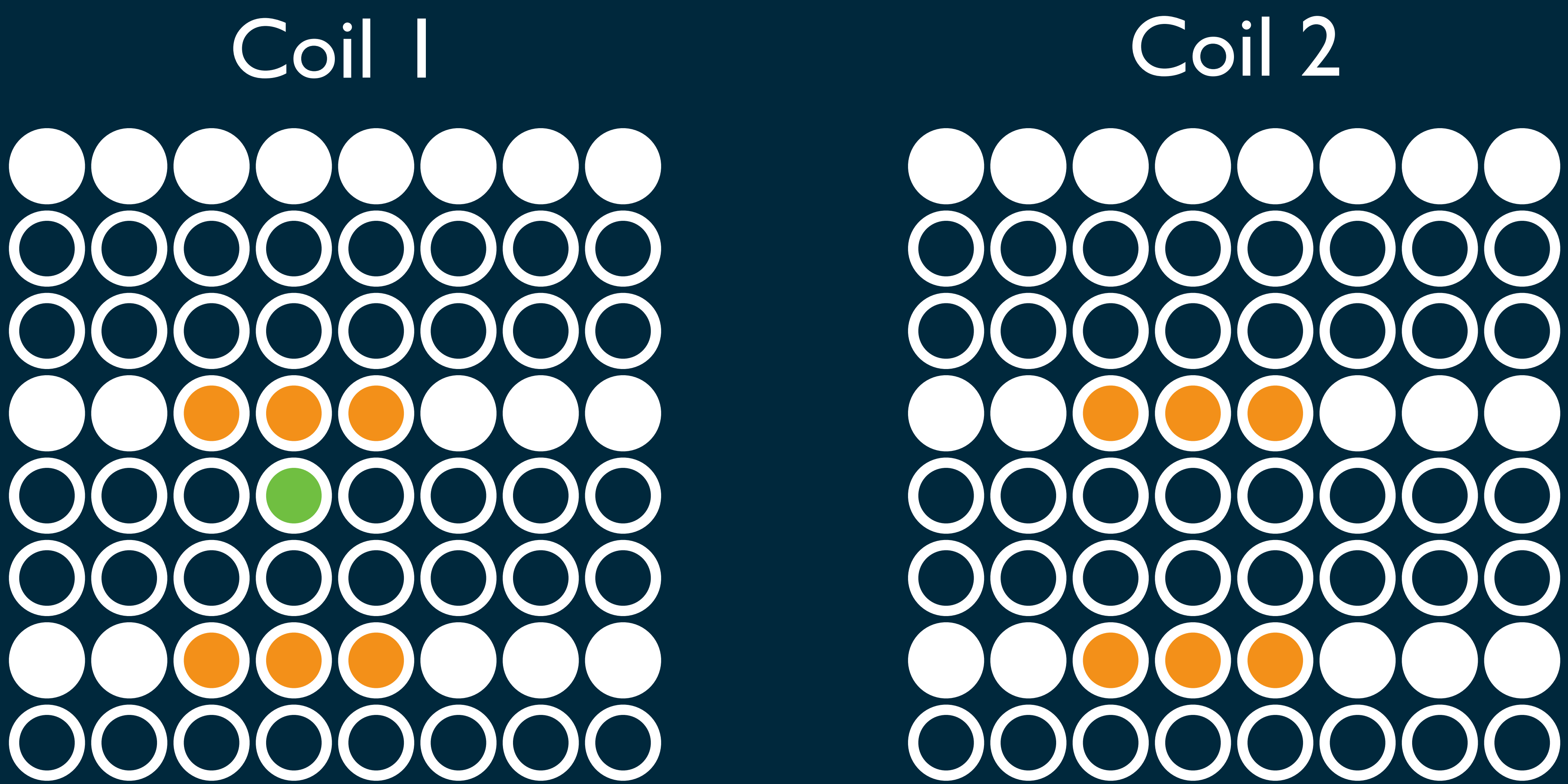
GRAPPA Geometry



Choose a kernel geometry

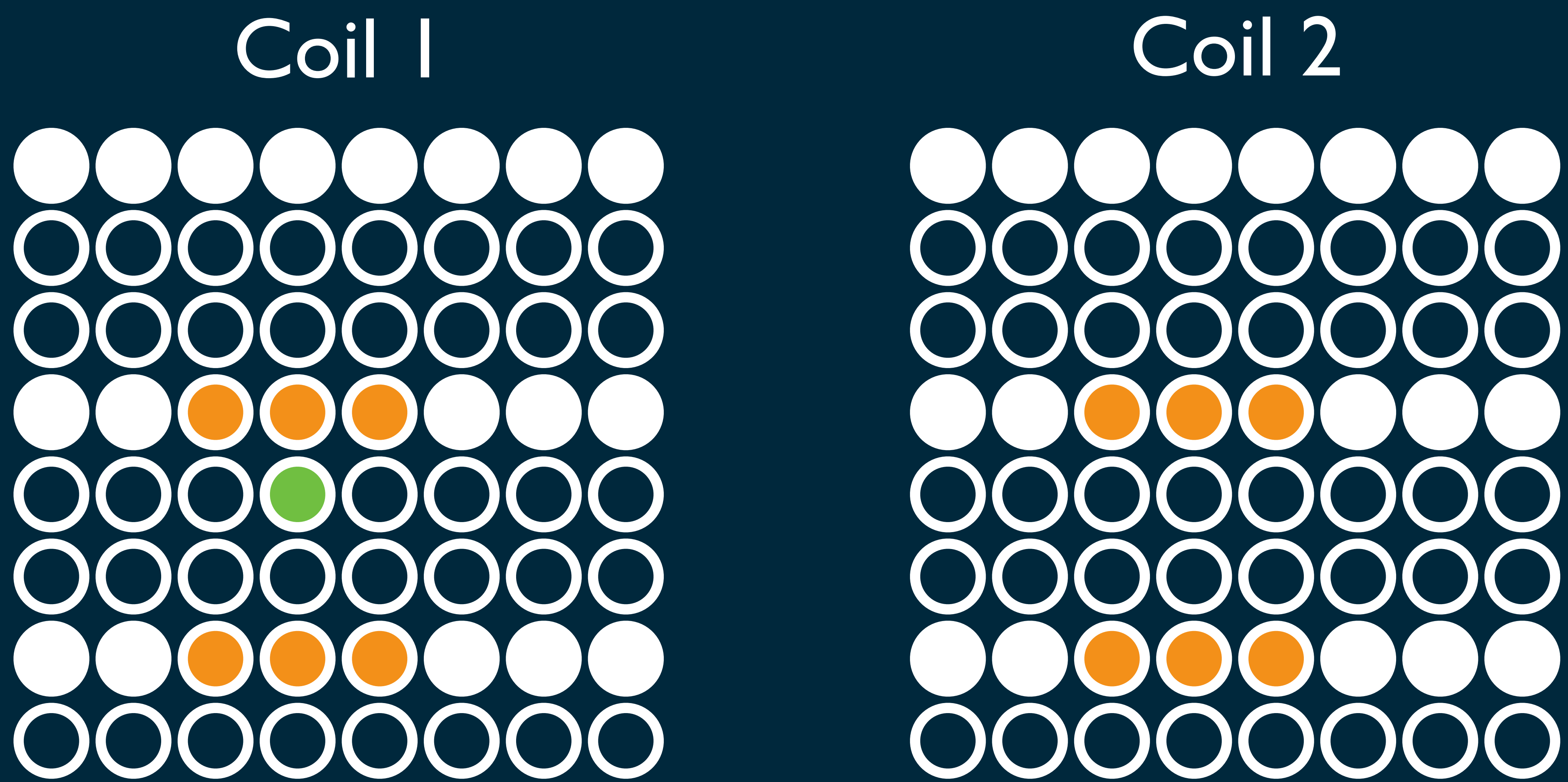
This dictates the spatial relationship between source and target points

GRAPPA Geometry



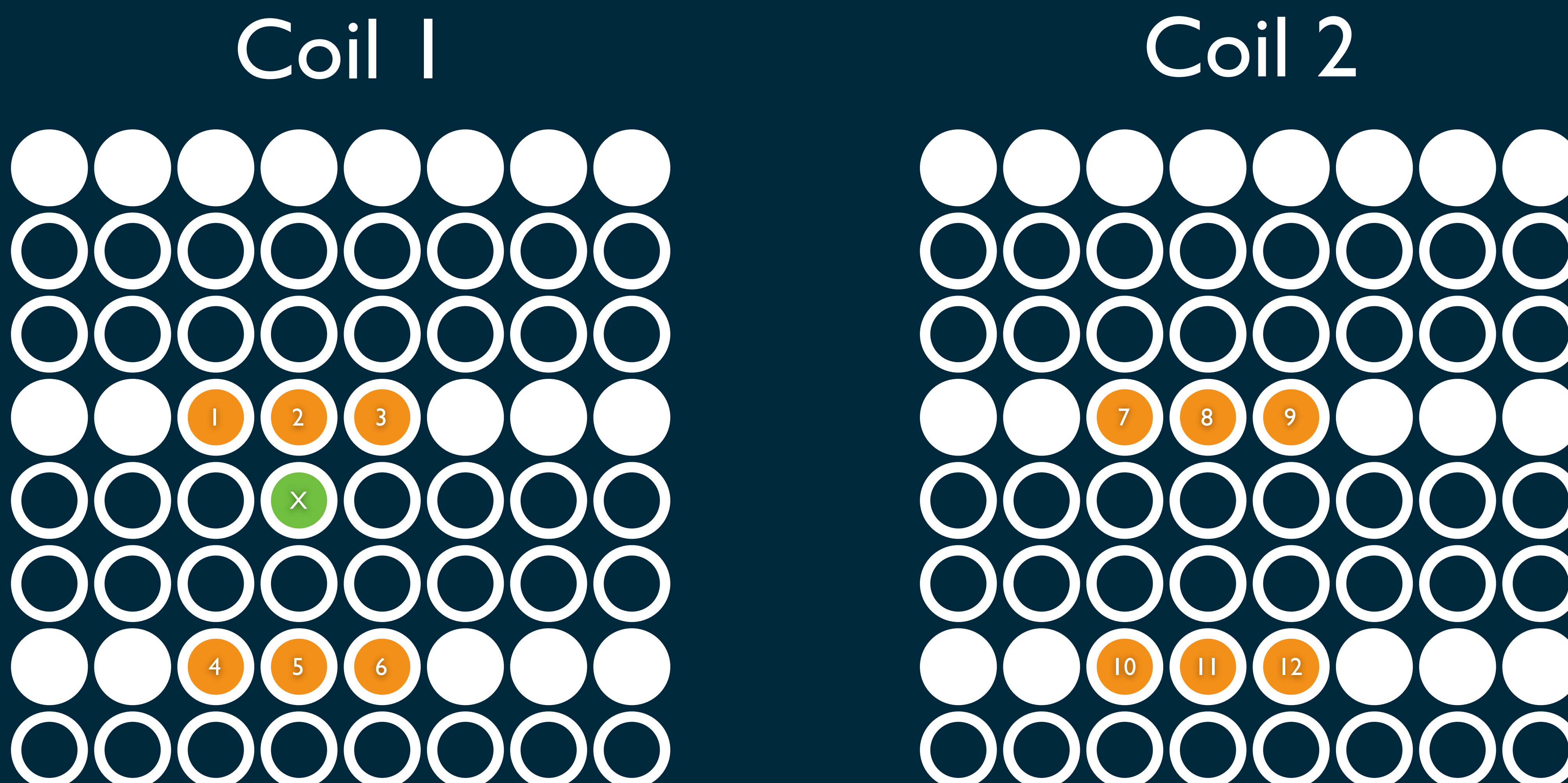
This is a [3,2] kernel geometry
3 points in x, 2 in y

GRAPPA Geometry



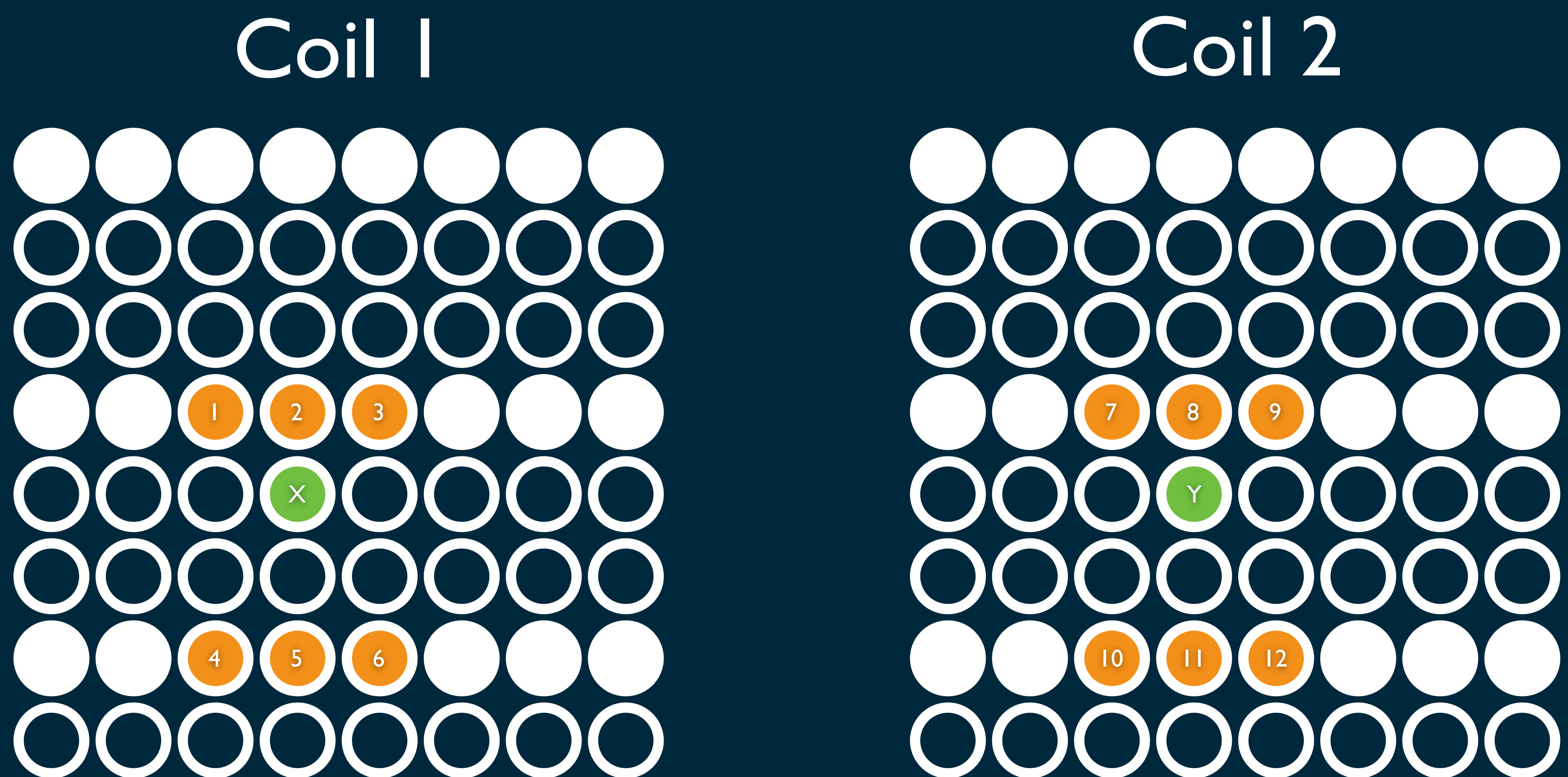
The kernel weights relate the source points from ALL coils to a target point in ONE coil

GRAPPA Geometry



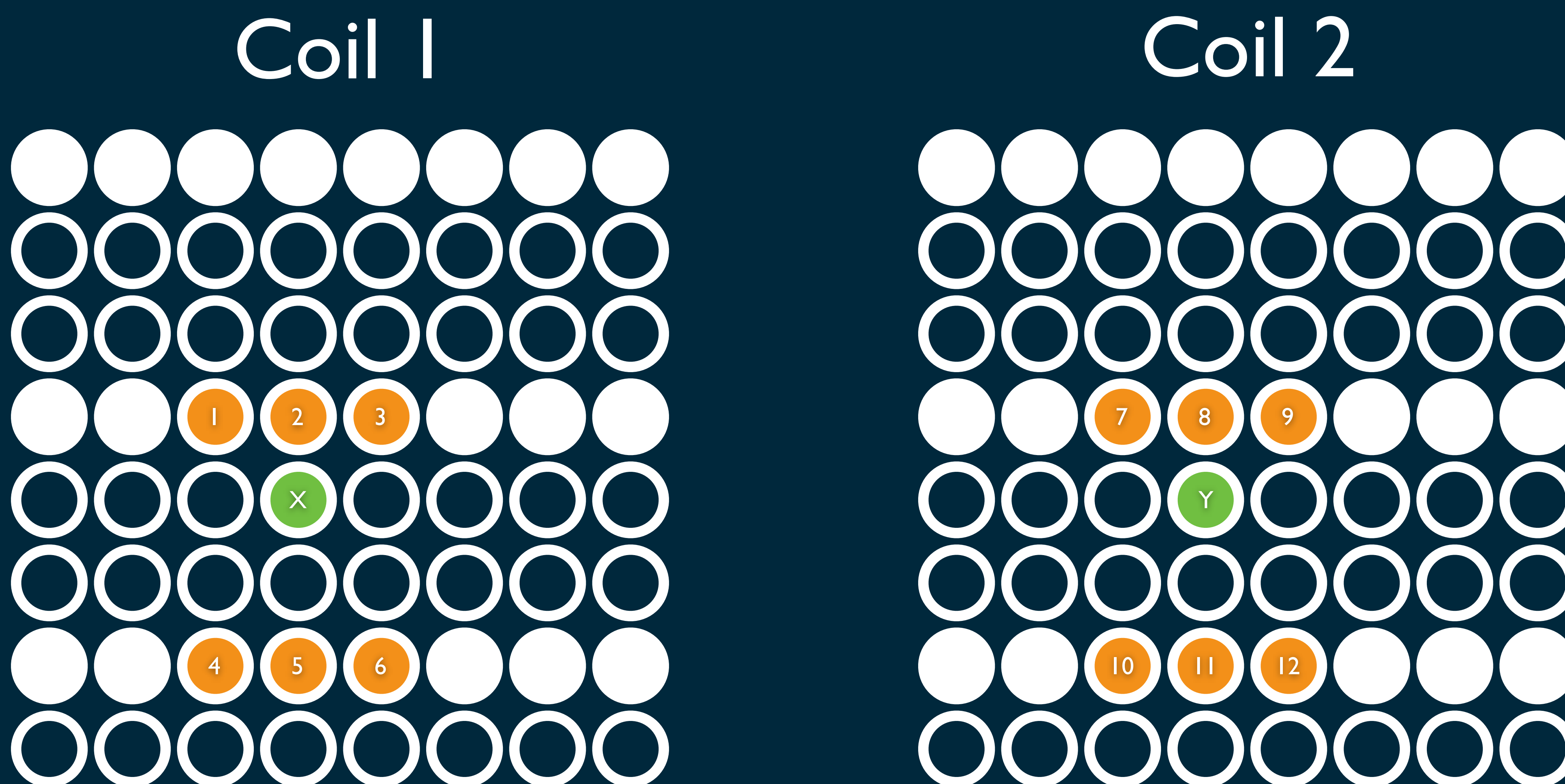
$$X = w_{1,1} \times 1 + w_{1,2} \times 2 + \ldots w_{1,12} \times 12$$

GRAPPA Geometry



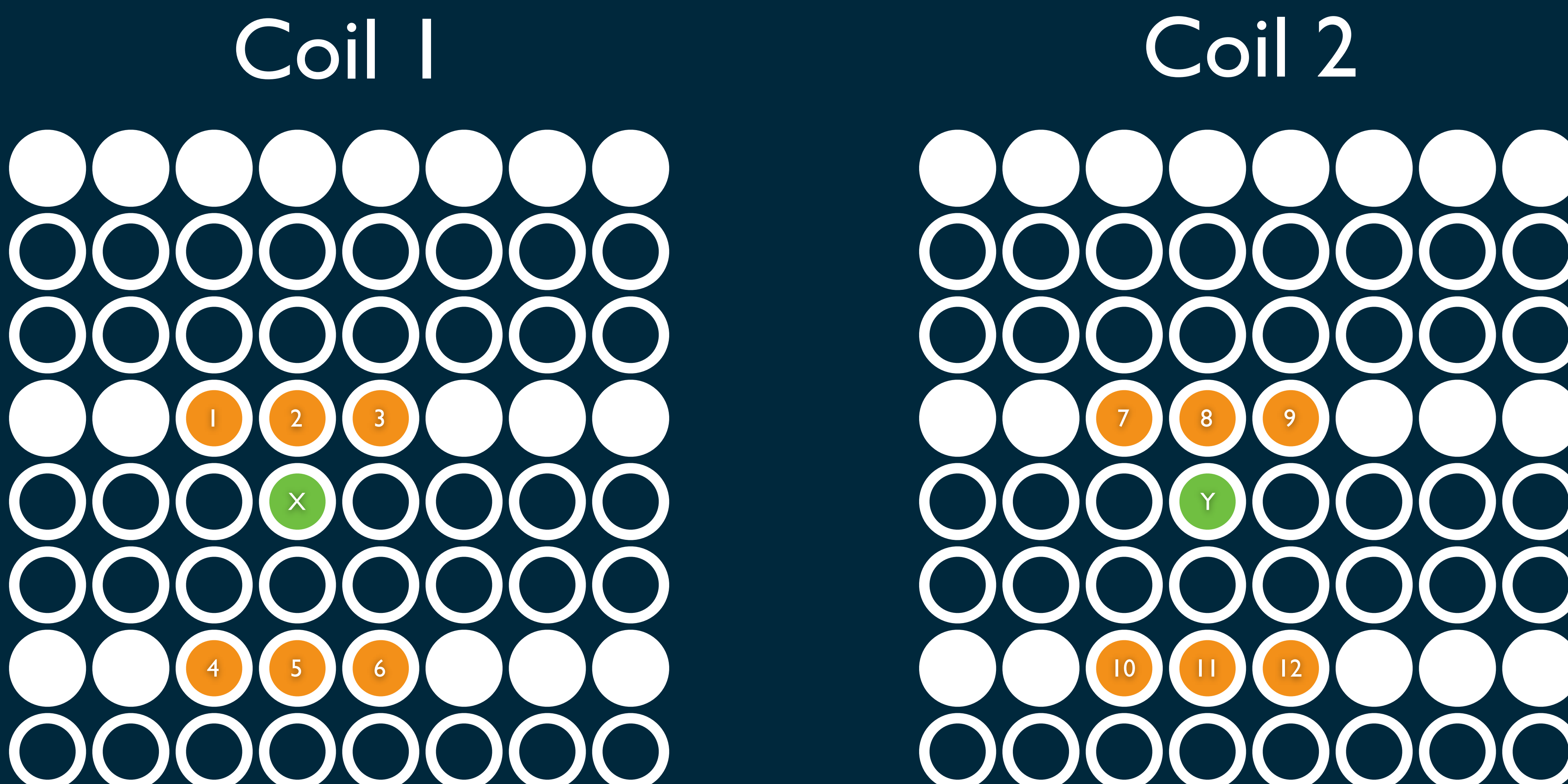
$$Y = w_{2,1} \times 1 + w_{2,2} \times 2 + \ldots w_{2,12} \times 12$$

GRAPPA Geometry



For this kernel, the all the weights can form a matrix
Each row corresponds to a different target coil
And one column per source point

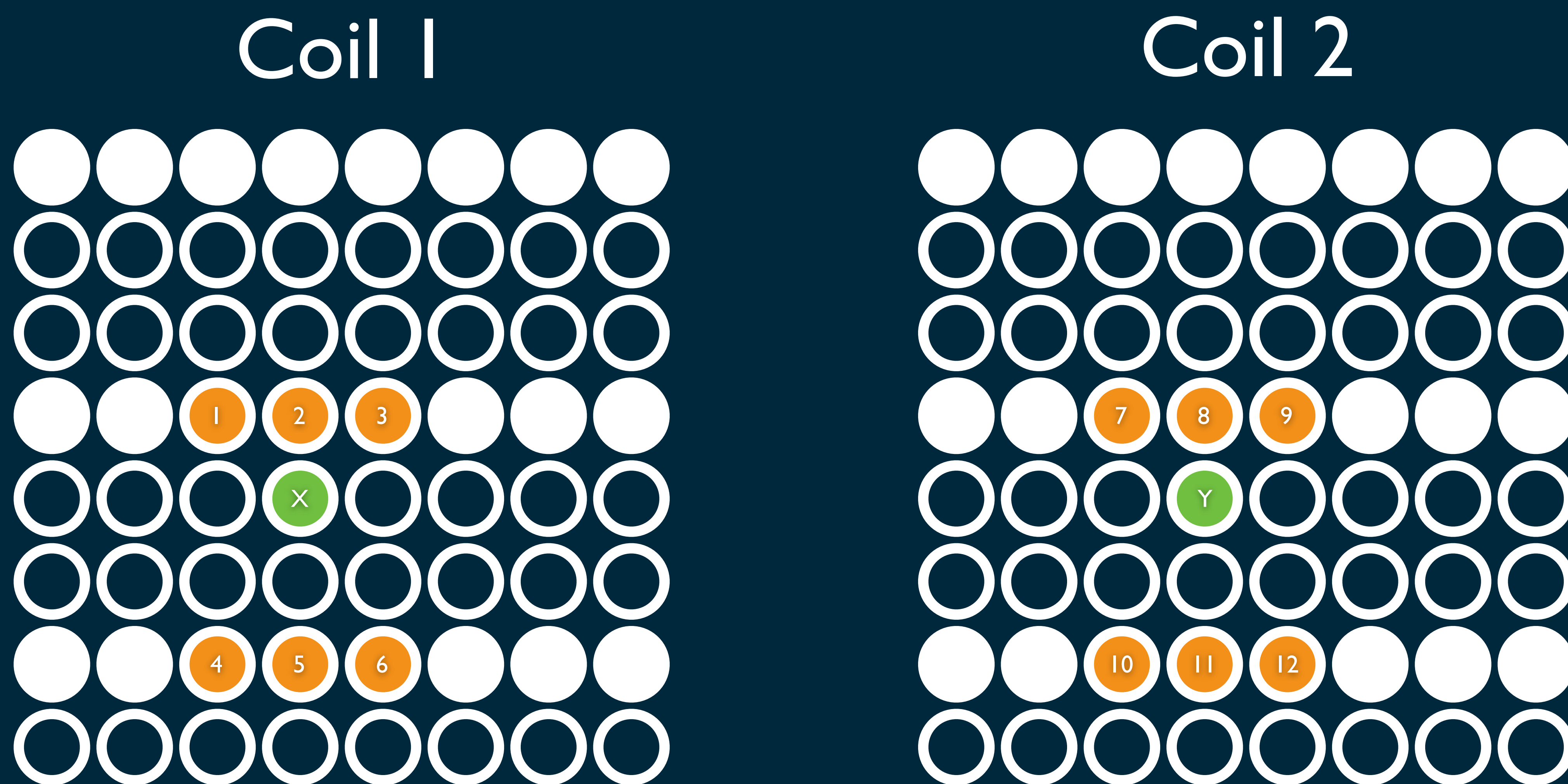
GRAPPA Geometry



$$X = w_{1,1} \times 1 + w_{1,2} \times 2 + \ldots w_{1,12} \times 12$$

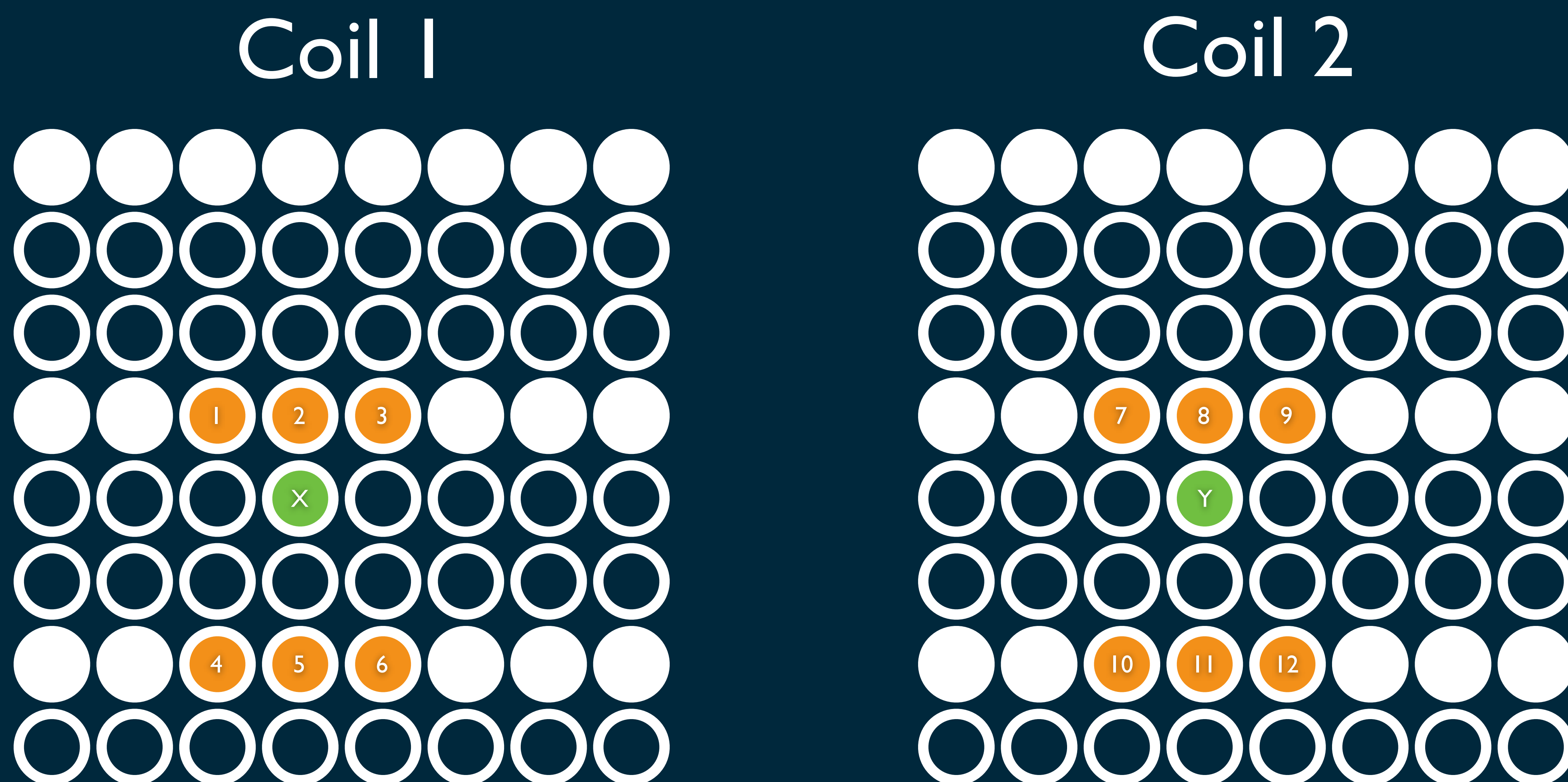
$$Y = w_{2,1} \times 1 + w_{2,2} \times 2 + \ldots w_{2,12} \times 12$$

GRAPPA Geometry



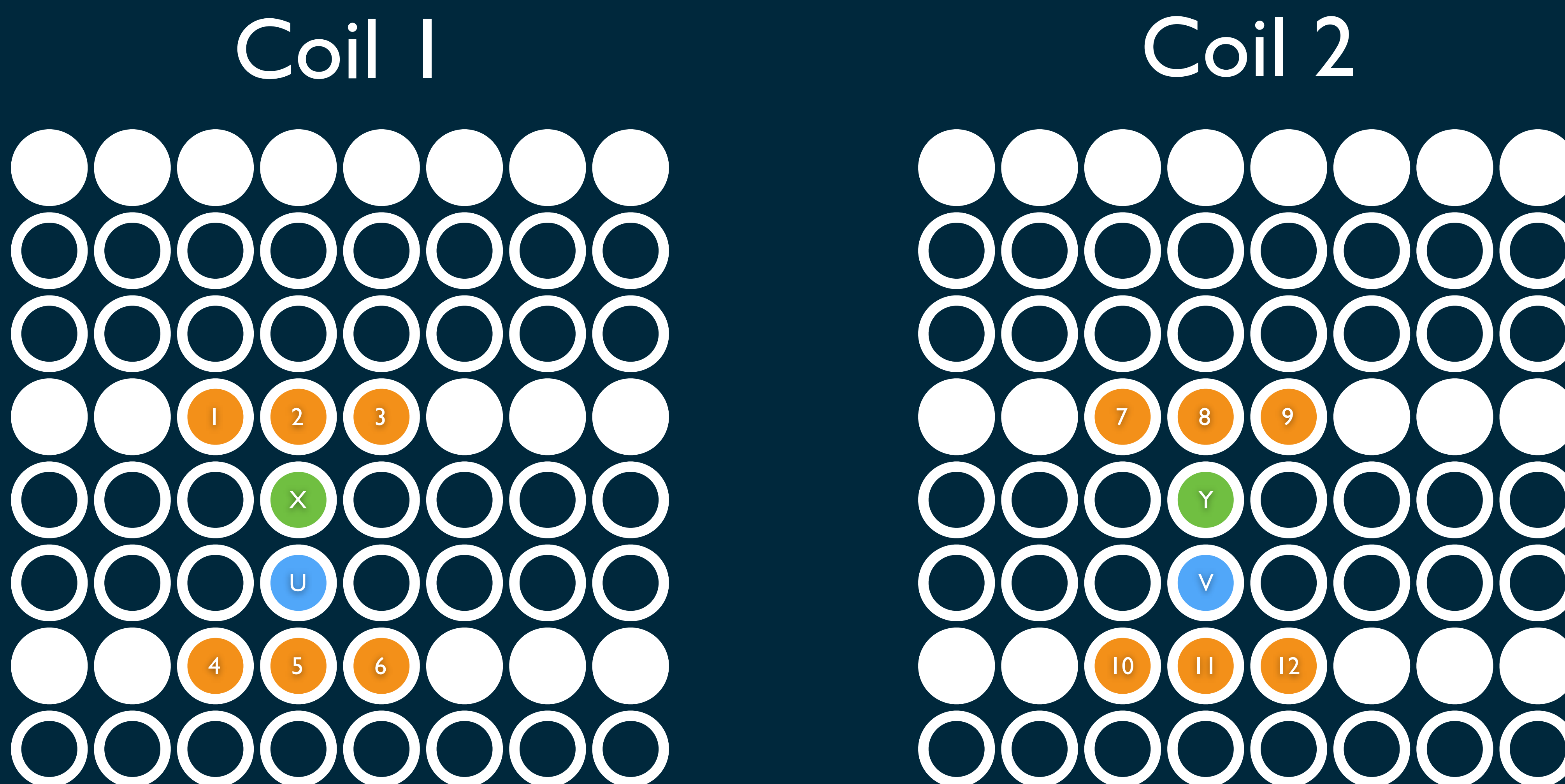
$$\begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} w_{1,1} & w_{1,2} & \cdots & w_{1,12} \\ w_{2,1} & w_{2,2} & \cdots & w_{2,12} \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ \vdots \\ 12 \end{bmatrix}$$

GRAPPA Geometry



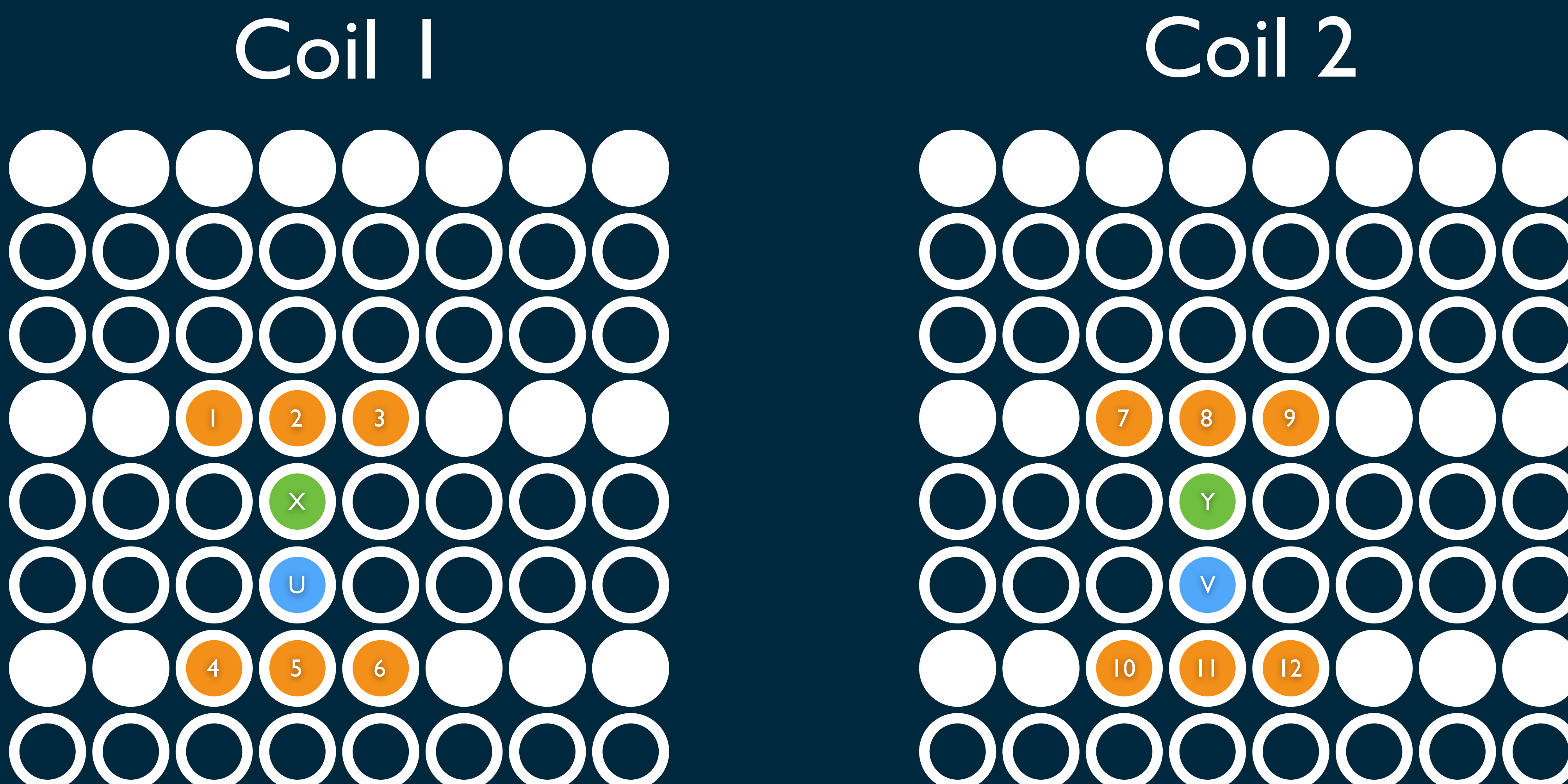
$$\begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} W^{(1)}{}_{1,1} & W^{(1)}{}_{1,2} & \cdots & W^{(1)}{}_{1,12} \\ W^{(1)}{}_{2,1} & W^{(1)}{}_{2,2} & \cdots & W^{(1)}{}_{2,12} \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ \vdots \\ 12 \end{bmatrix}$$

GRAPPA Geometry



$$\begin{bmatrix} U \\ V \end{bmatrix} = \begin{bmatrix} W^{(2)}{}_{1,1} & W^{(2)}{}_{1,2} & \cdots & W^{(2)}{}_{1,12} \\ W^{(2)}{}_{2,1} & W^{(2)}{}_{2,2} & \cdots & W^{(2)}{}_{2,12} \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ \vdots \\ 12 \end{bmatrix}$$

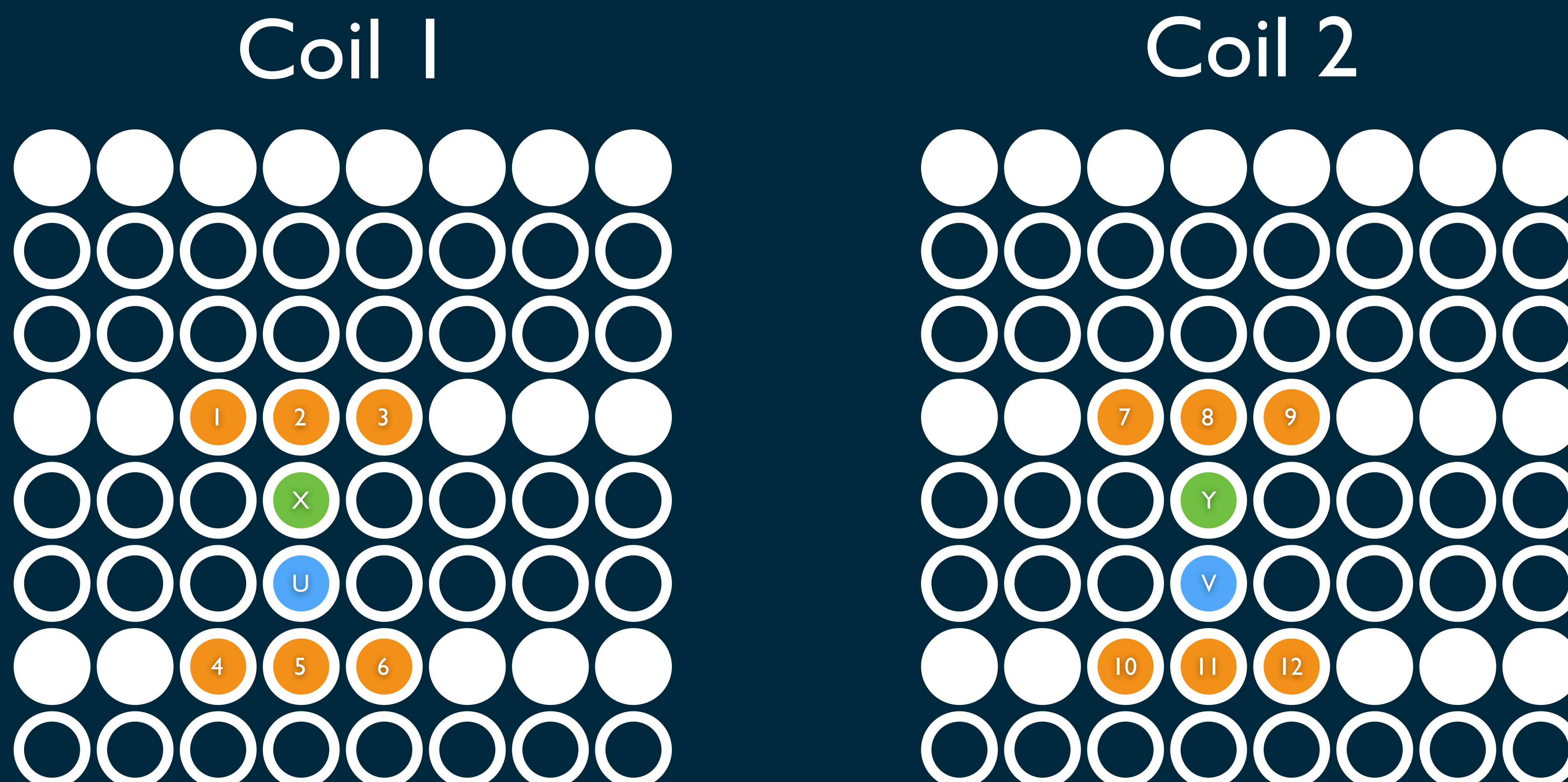
GRAPPA Geometry



$$\begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} W^{(1)}_{1,1} & W^{(1)}_{1,2} & \cdots & W^{(1)}_{1,12} \\ W^{(1)}_{2,1} & W^{(1)}_{2,2} & \cdots & W^{(1)}_{2,12} \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ \vdots \\ 12 \end{bmatrix}$$

$$\begin{bmatrix} U \\ V \end{bmatrix} = \begin{bmatrix} W^{(2)}_{1,1} & W^{(2)}_{1,2} & \cdots & W^{(2)}_{1,12} \\ W^{(2)}_{2,1} & W^{(2)}_{2,2} & \cdots & W^{(2)}_{2,12} \end{bmatrix}$$

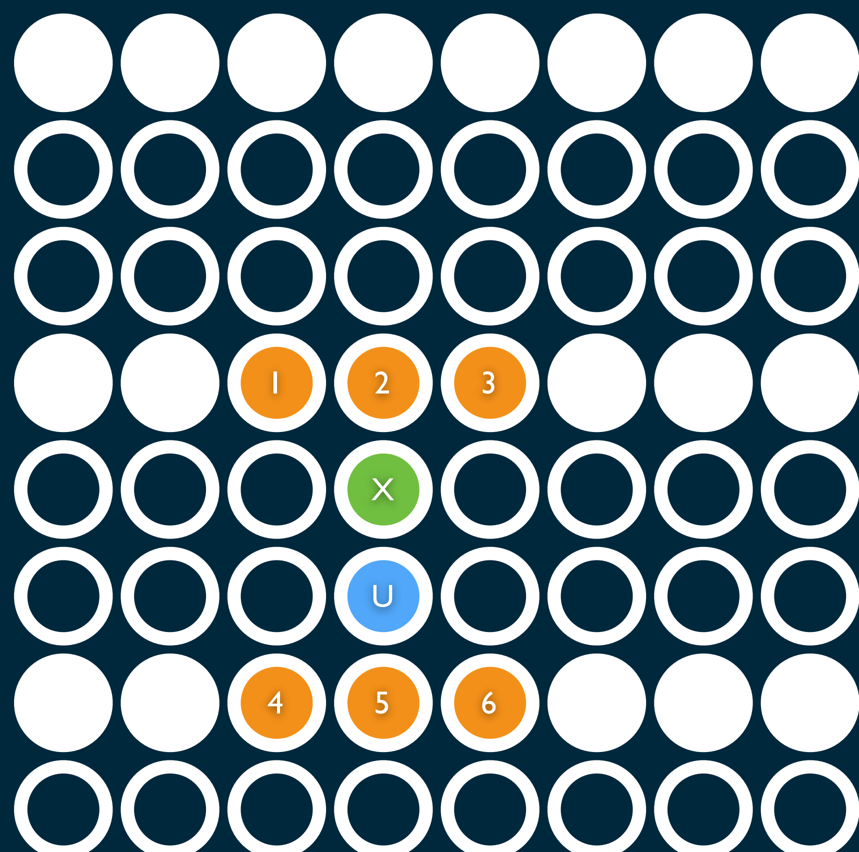
GRAPPA Geometry



$$\begin{bmatrix} X \\ Y \\ U \\ V \end{bmatrix} = \begin{bmatrix} W^{(1)}_{1,1} & W^{(1)}_{1,2} & \cdots & W^{(1)}_{1,12} \\ W^{(1)}_{2,1} & W^{(1)}_{2,2} & \cdots & W^{(1)}_{2,12} \\ W^{(2)}_{1,1} & W^{(2)}_{1,2} & \cdots & W^{(2)}_{1,12} \\ W^{(2)}_{2,1} & W^{(2)}_{2,2} & \cdots & W^{(2)}_{2,12} \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ \vdots \\ 12 \end{bmatrix}$$

GRAPPA Geometry

Coil 1



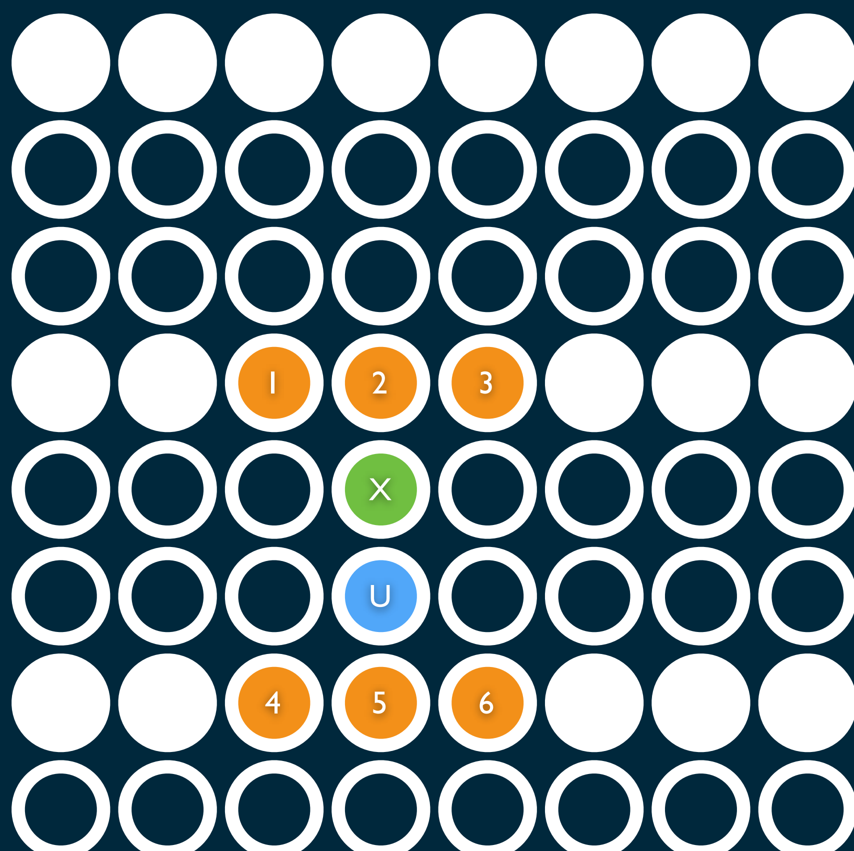
In general, there are R-1 different kernel types

Every missing point should fit into one of these classes

$$\begin{bmatrix} X \\ Y \\ U \\ V \end{bmatrix} = \begin{bmatrix} W^{(1)}_{1,1} & W^{(1)}_{1,2} & \cdots & W^{(1)}_{1,12} \\ W^{(1)}_{2,1} & W^{(1)}_{2,2} & \cdots & W^{(1)}_{2,12} \\ W^{(2)}_{1,1} & W^{(2)}_{1,2} & \cdots & W^{(2)}_{1,12} \\ W^{(2)}_{2,1} & W^{(2)}_{2,2} & \cdots & W^{(2)}_{2,12} \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ \vdots \\ 12 \end{bmatrix}$$

GRAPPA Geometry

Coil 1



So in total, for a system with L coils, kernel size [kx, ky], and undersampling factor R

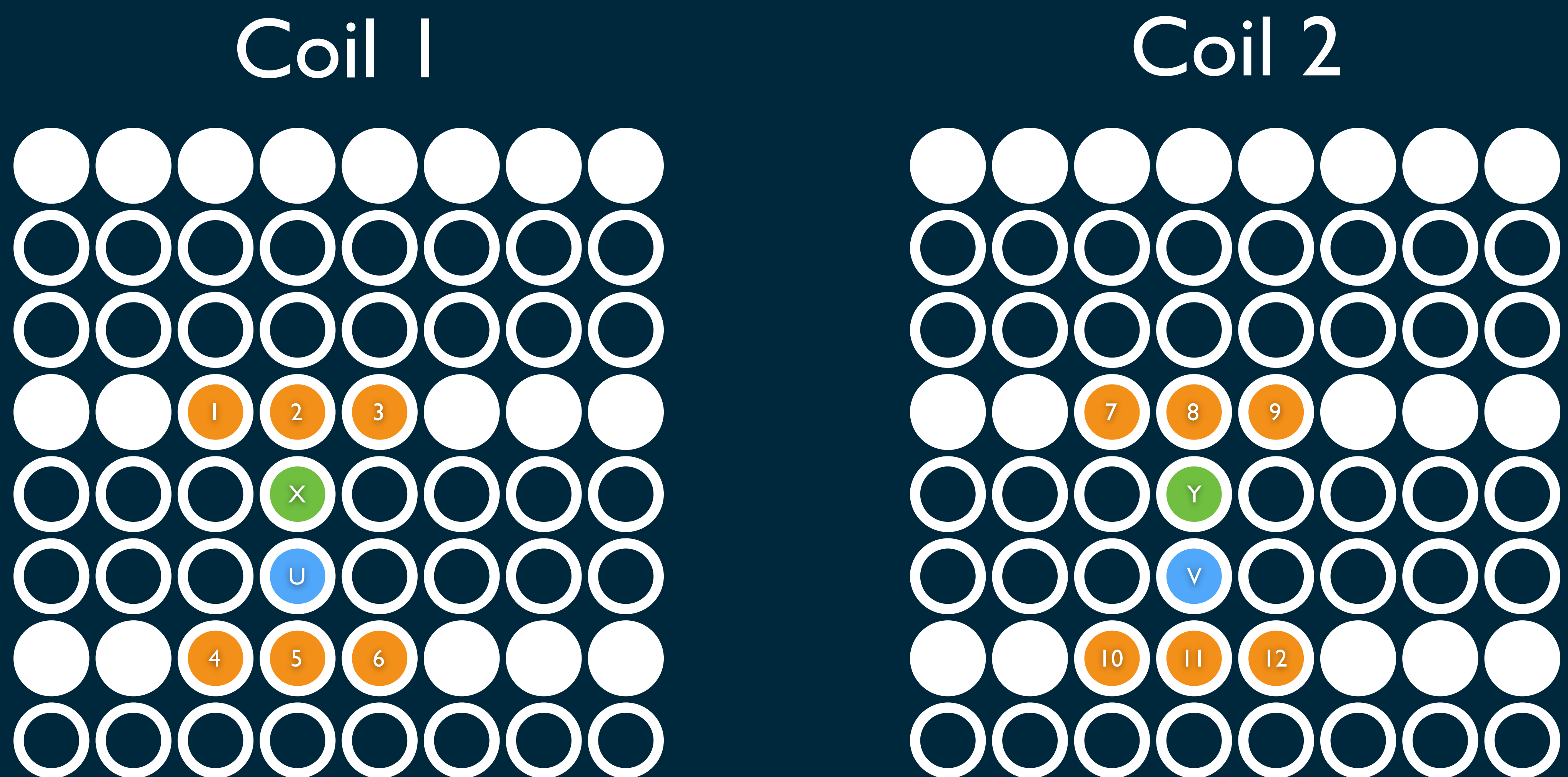
You need an :

`L*(R-1) by L*kx*ky`

matrix of weights

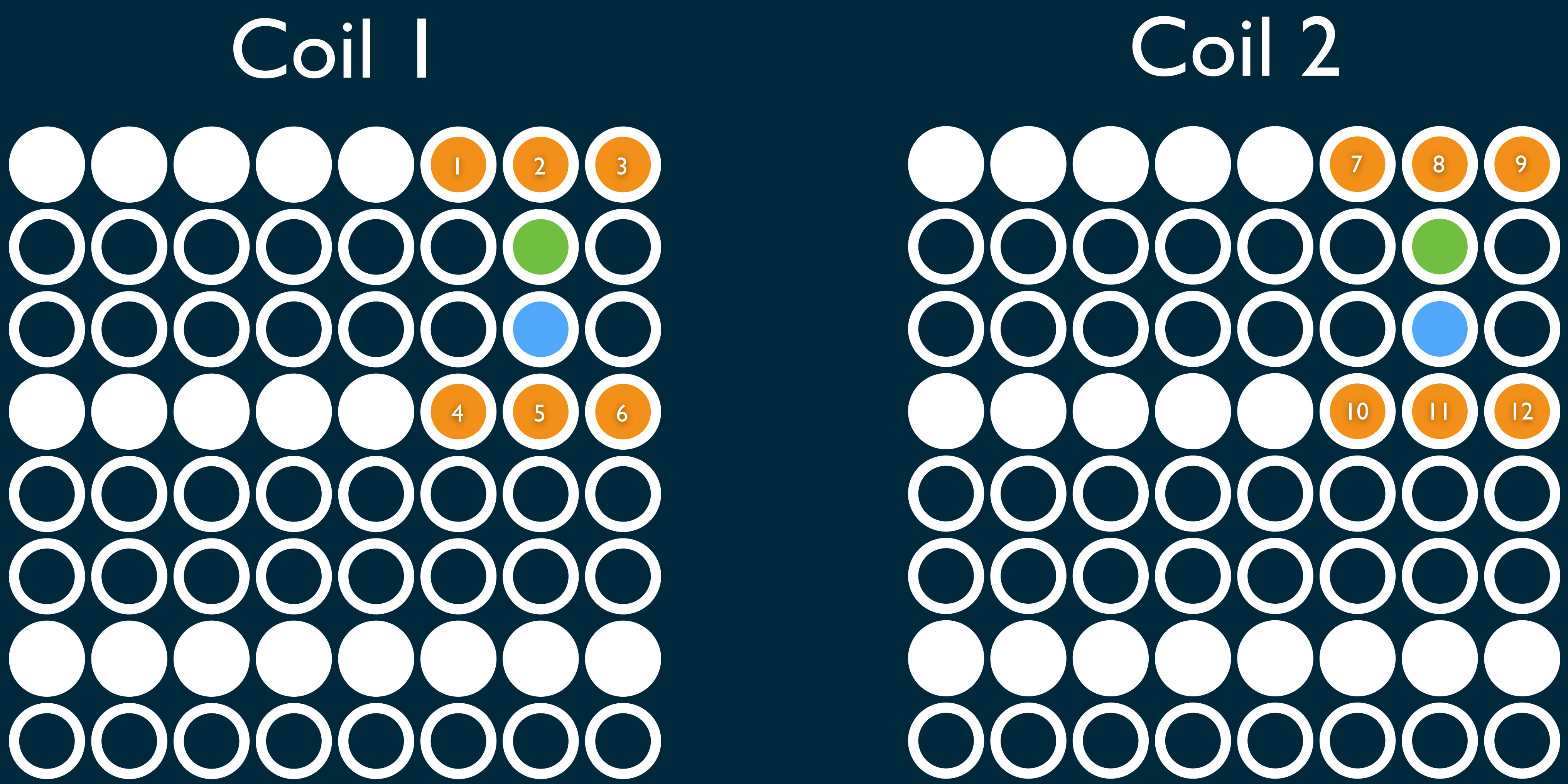
$$\begin{bmatrix} X \\ Y \\ U \\ V \end{bmatrix} = \begin{bmatrix} w^{(1)}_{1,1} & w^{(1)}_{1,2} & \cdots & w^{(1)}_{1,12} \\ w^{(1)}_{2,1} & w^{(1)}_{2,2} & \cdots & w^{(1)}_{2,12} \\ w^{(2)}_{1,1} & w^{(2)}_{1,2} & \cdots & w^{(2)}_{1,12} \\ w^{(2)}_{2,1} & w^{(2)}_{2,2} & \cdots & w^{(2)}_{2,12} \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ \vdots \\ 12 \end{bmatrix}$$

GRAPPA Geometry



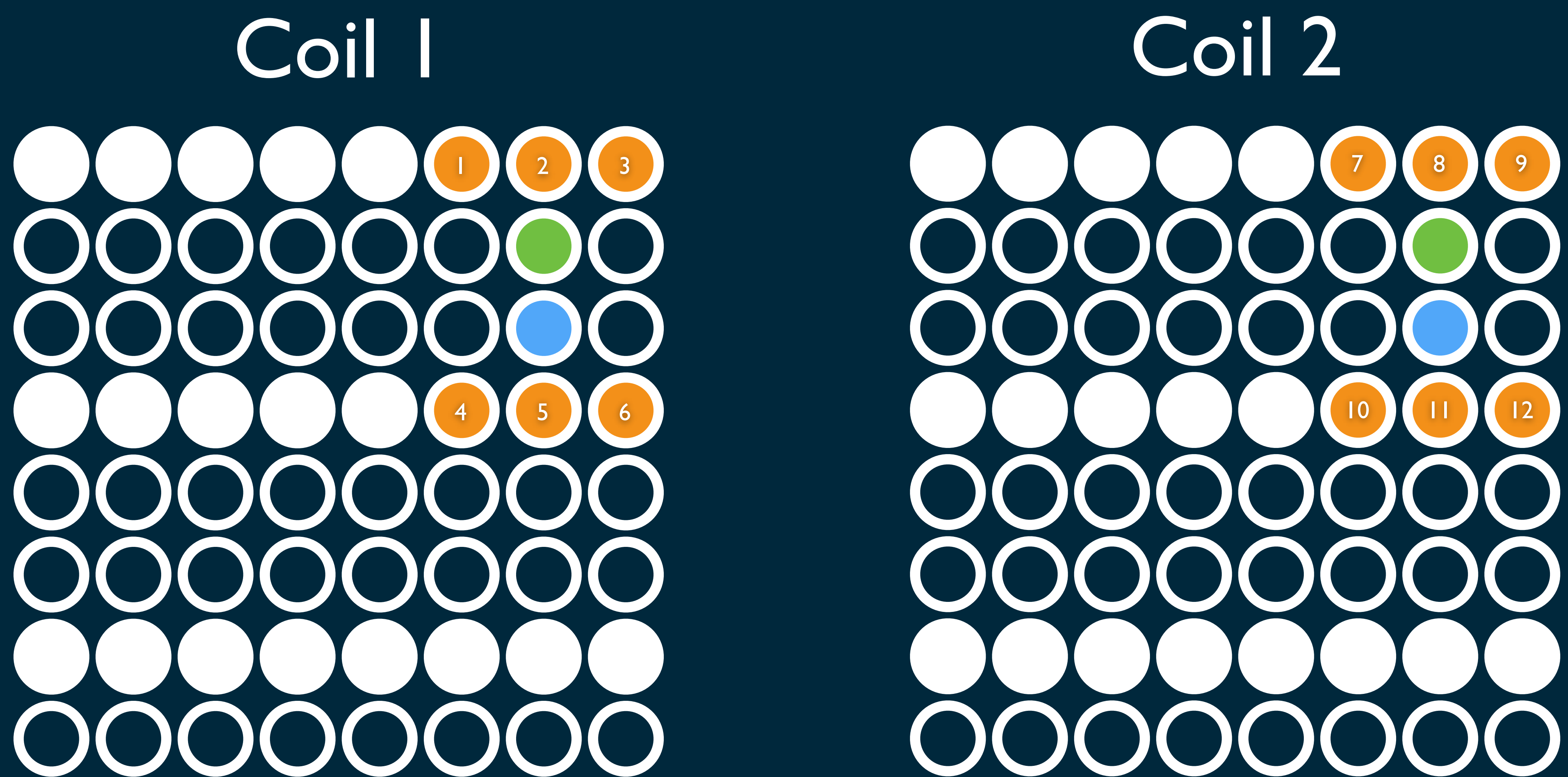
You solve for all the missing points by applying the same relative kernel weights

GRAPPA Geometry



You solve for all the missing points by applying the same relative kernel weights

GRAPPA Geometry



They way you find the weights, is that you solve the exact same problem

GRAPPA Geometry

$$
\begin{array}{ccccc}
\text{target} & & \text{weights} & & \text{source} \\
\begin{bmatrix} X \\ Y \\ U \\ V \end{bmatrix} & = &
\begin{bmatrix}
w^{(1)}_{1,1} & w^{(1)}_{1,2} & \cdots & w^{(1)}_{1,12} \\
w^{(1)}_{2,1} & w^{(1)}_{2,2} & \cdots & w^{(1)}_{2,12} \\
w^{(2)}_{1,1} & w^{(2)}_{1,2} & \cdots & w^{(2)}_{1,12} \\
w^{(2)}_{2,1} & w^{(2)}_{2,2} & \cdots & w^{(2)}_{2,12}
\end{bmatrix} & &
\begin{bmatrix} 1 \\ 2 \\ \vdots \\ 12 \end{bmatrix}
\end{array}
$$

They way you find the weights, is that you solve the exact same problem

But now, instead of the GRAPPA synthesis problem where x, y, u, v are unknown

GRAPPA Geometry

$$\begin{array}{cccc} \text{target} & & \text{weights} & \text{source} \\ \begin{bmatrix} X \\ Y \\ U \\ V \end{bmatrix} & = & \begin{bmatrix} w^{(1)}_{1,1} & w^{(1)}_{1,2} & \cdots & w^{(1)}_{1,12} \\ w^{(1)}_{2,1} & w^{(1)}_{2,2} & \cdots & w^{(1)}_{2,12} \\ w^{(2)}_{1,1} & w^{(2)}_{1,2} & \cdots & w^{(2)}_{1,12} \\ w^{(2)}_{2,1} & w^{(2)}_{2,2} & \cdots & w^{(2)}_{2,12} \end{bmatrix} & \begin{bmatrix} 1 \\ 2 \\ \vdots \\ 12 \end{bmatrix} \end{array}$$

You require some fully sampled calibration data

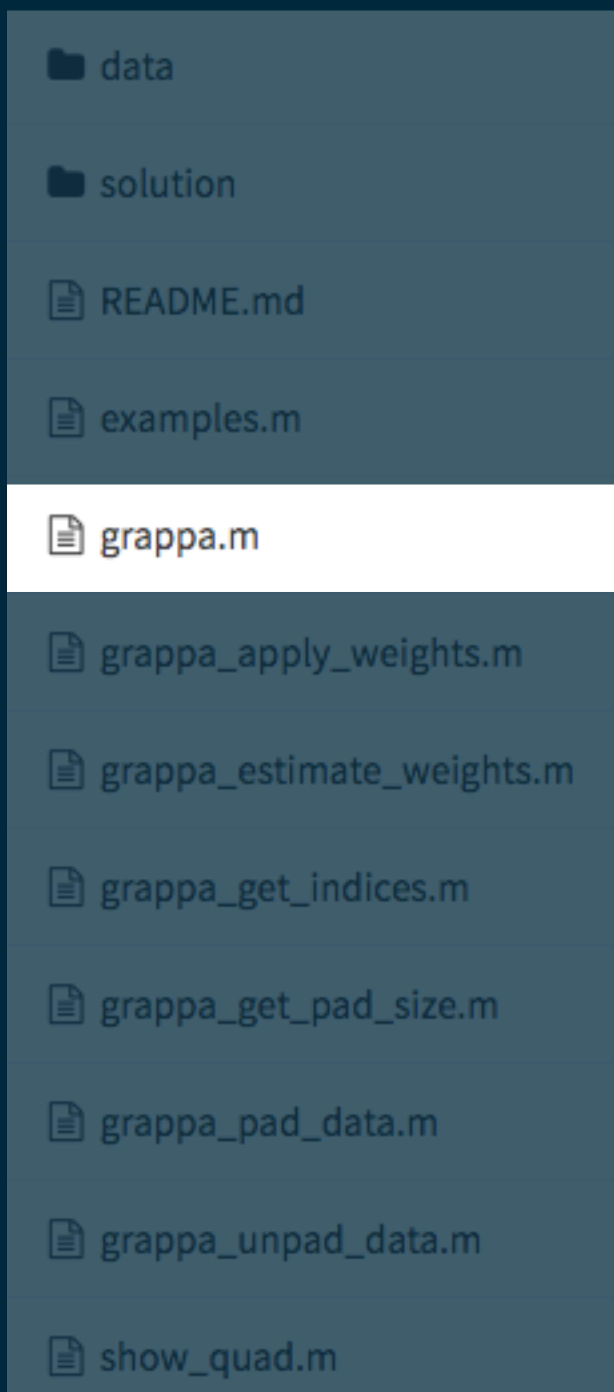
Where x, y, u, v are known, and you solve for w
The shift-invariance of the kernel weights
allows you to fit this over a large set of different
source and target points

GRAPPA Practical

- `https://psg.fmrib.ox.ac.uk/mchiew/grappa-practical/`
- `git clone https://psg.fmrib.ox.ac.uk/mchiew/grappa-practical.git`
- `https://psg.fmrib.ox.ac.uk/mchiew/grappa-practical/repository/archive.zip`

GRAPPA Practical

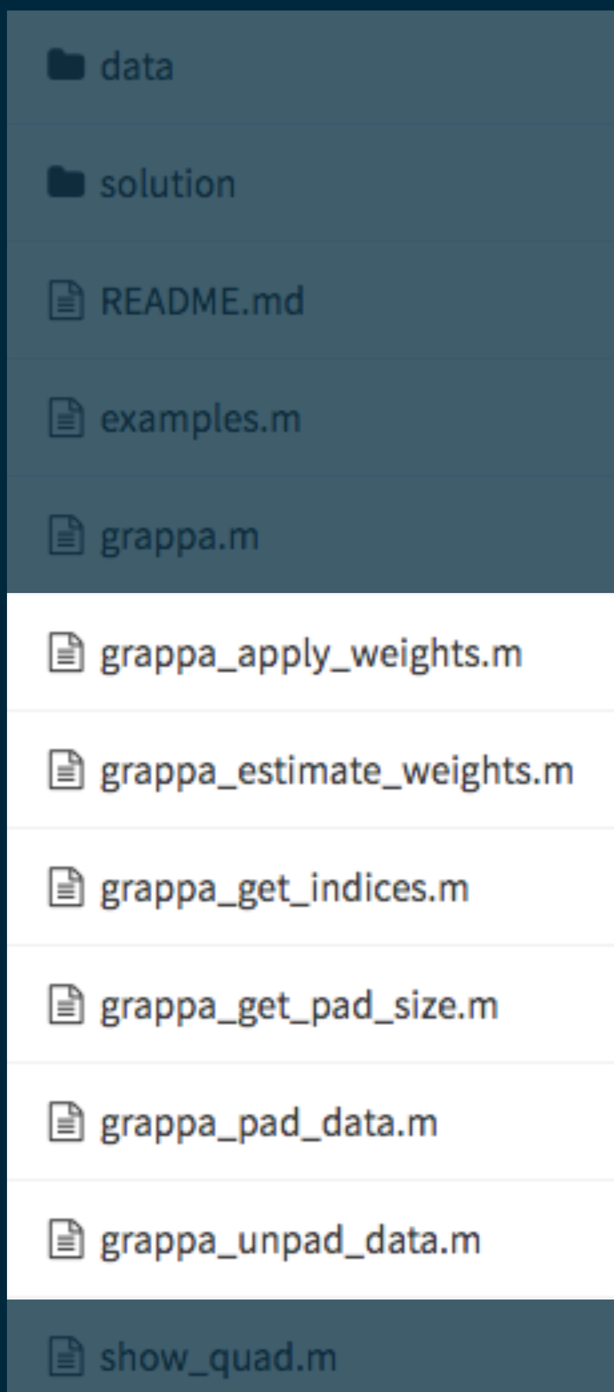
- `git clone https://psg.fmrib.ox.ac.uk/mchiew/grappa-practical.git`
- `https://psg.fmrib.ox.ac.uk/mchiew/grappa-practical/repository/archive.zip`



- grappa.m is the main GRAPPA function
- It calls 6 different helper functions

GRAPPA Practical

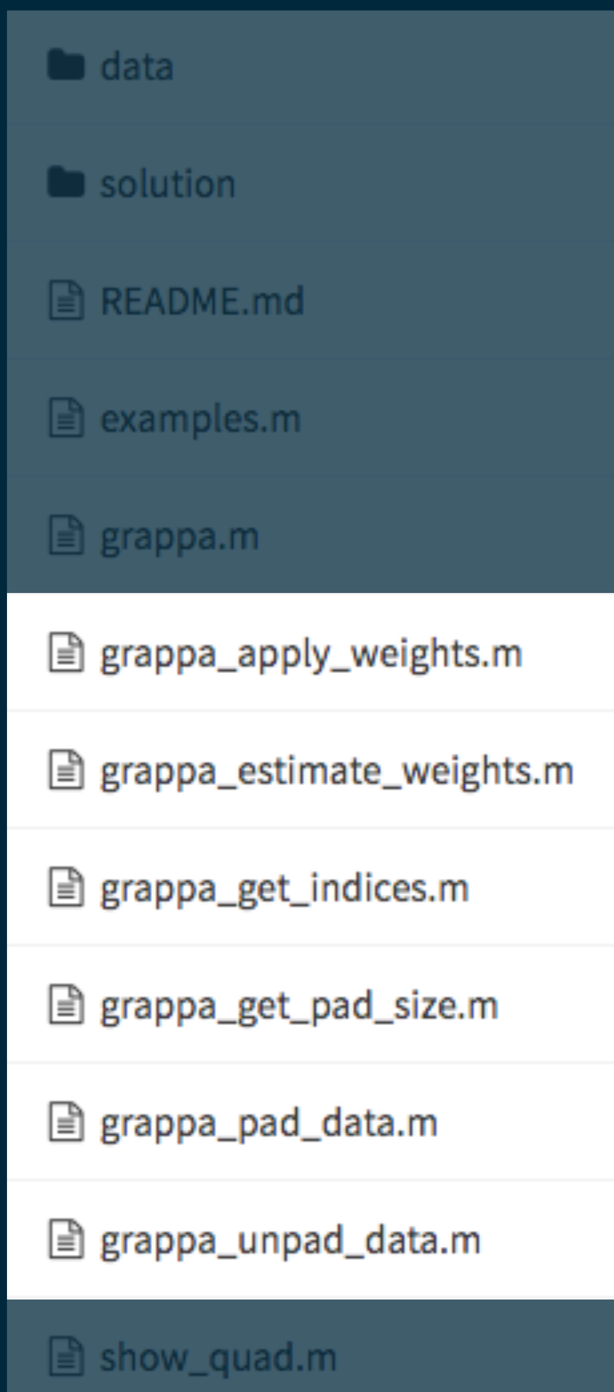
- `git clone https://psg.fmrib.ox.ac.uk/mchiew/grappa-practical.git`
- `https://psg.fmrib.ox.ac.uk/mchiew/grappa-practical/repository/archive.zip`



- These functions contain all the GRAPPA logic
- grappa_{apply_weights.m, estimate_weights.m} are the actual interesting bits
- The rest are housekeeping

GRAPPA Practical

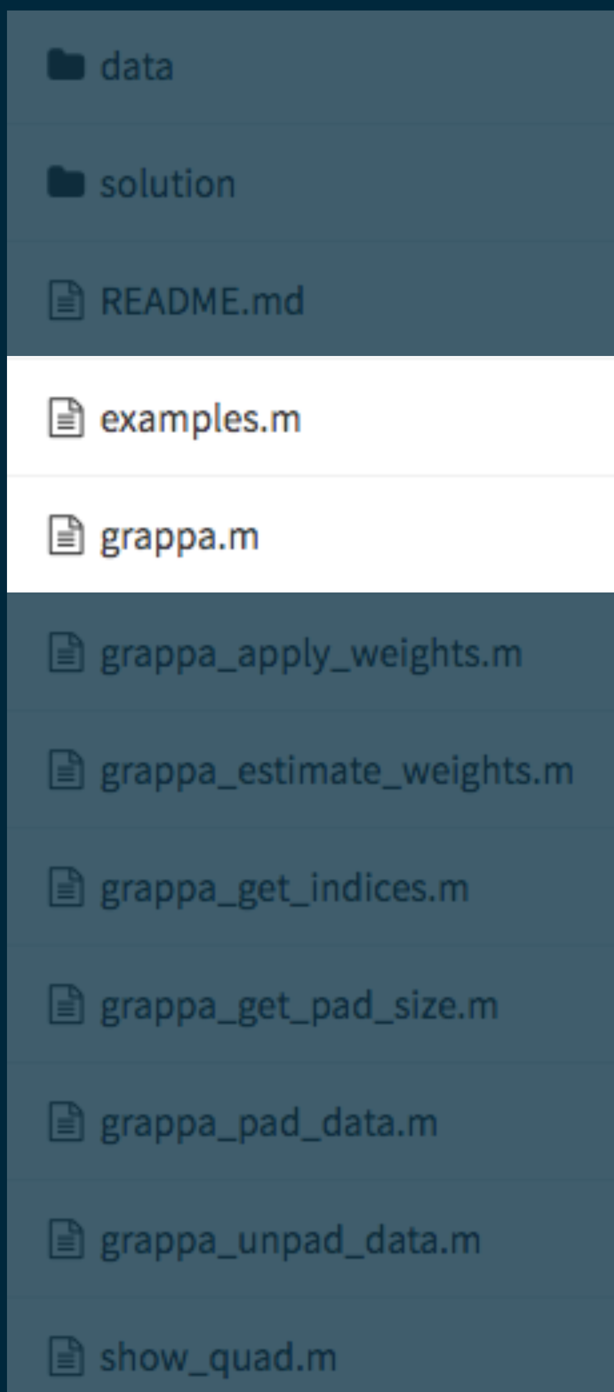
- `git clone https://psg.fmrib.ox.ac.uk/mchiew/grappa-practical.git`
- `https://psg.fmrib.ox.ac.uk/mchiew/grappa-practical/repository/archive.zip`



- Your job is to implement these internal bits
- The README has more detailed instructions on each component
- The files have function definitions and comments that should help you

GRAPPA Practical

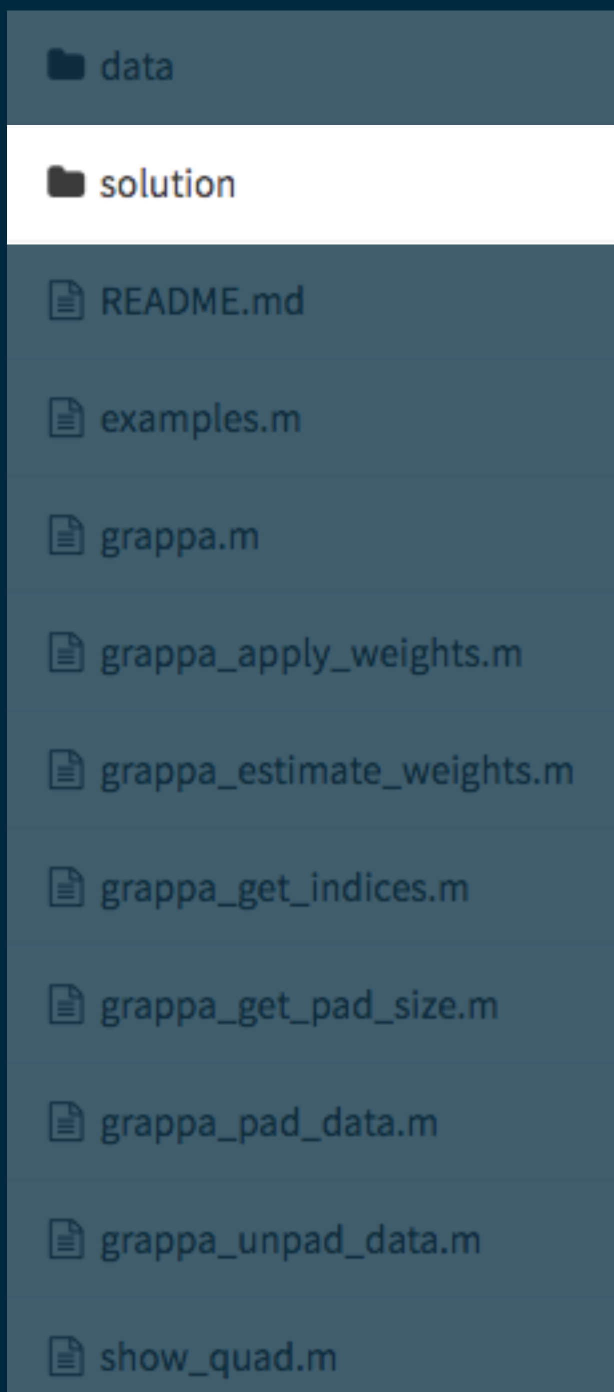
- `git clone https://psg.fmrib.ox.ac.uk/mchiew/grappa-practical.git`
- `https://psg.fmrib.ox.ac.uk/mchiew/grappa-practical/repository/archive.zip`



- Your goal is to be able to run grappa.m
- You can test this by running the examples.m script, which evaluates a few test cases

GRAPPA Practical

- `git clone https://psg.fmrib.ox.ac.uk/mchiew/grappa-practical.git`
- `https://psg.fmrib.ox.ac.uk/mchiew/grappa-practical/repository/archive.zip`



- A full working implementation for all the files can be found in the solution folder
- If you want to focus on the conceptually interesting aspects of GRAPPA, work on:
  - grappa_estimate_weights.m
  - grappa_apply_weights.m
- Copy the remaining files from solution