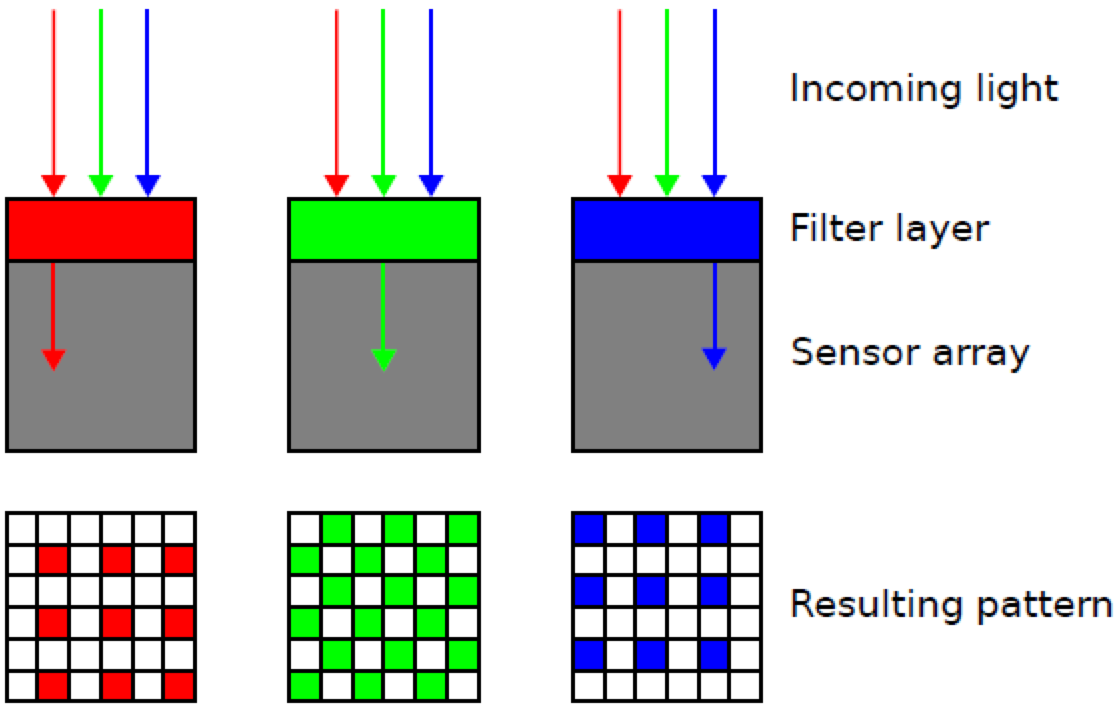
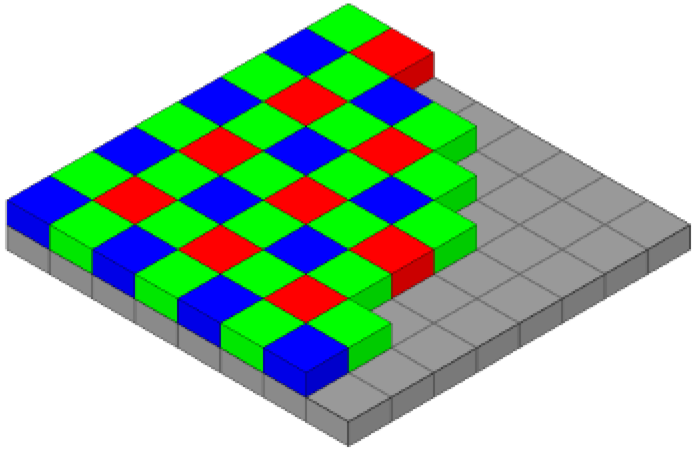
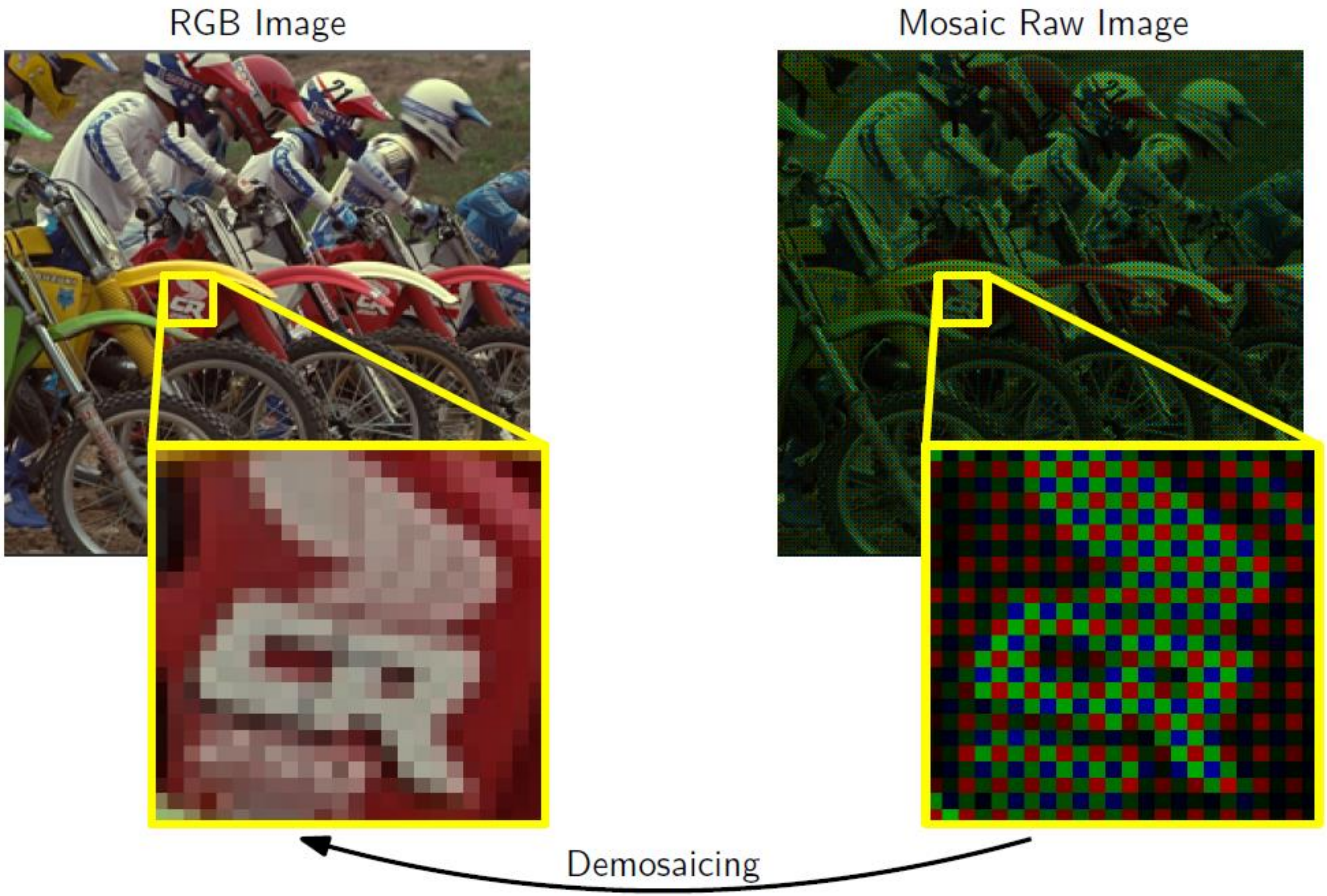
Instructions on Project 3

Color Demosaicing for Digital Cameras with Linear Regression

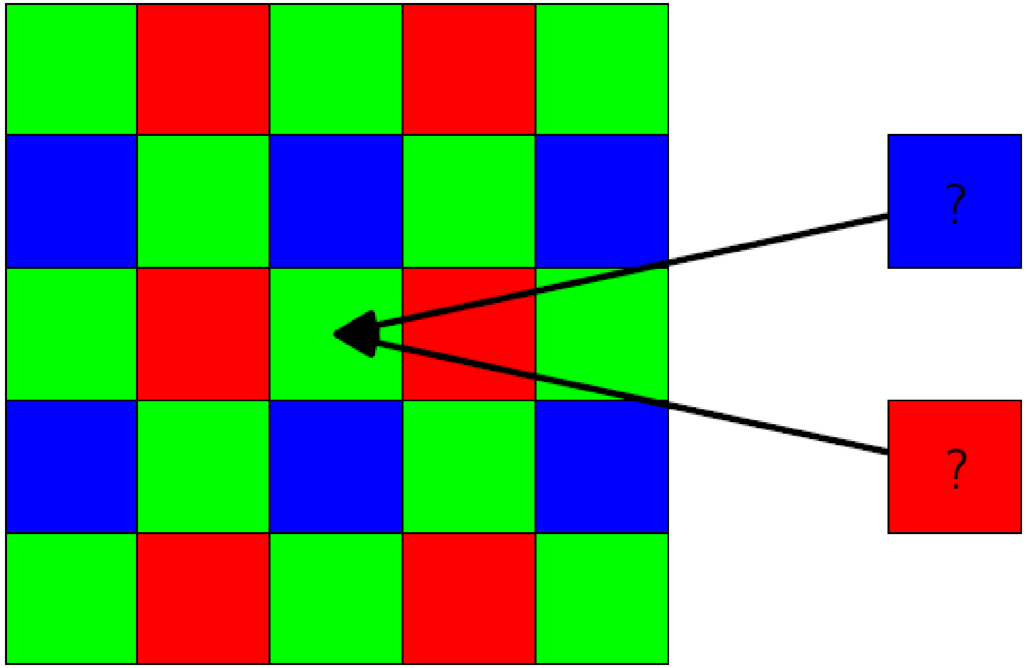
- Photo sensors cannot differentiate colours
- How to achieve colour imaging
  - Three-sensor camera
  - Colour filter array (CFA)

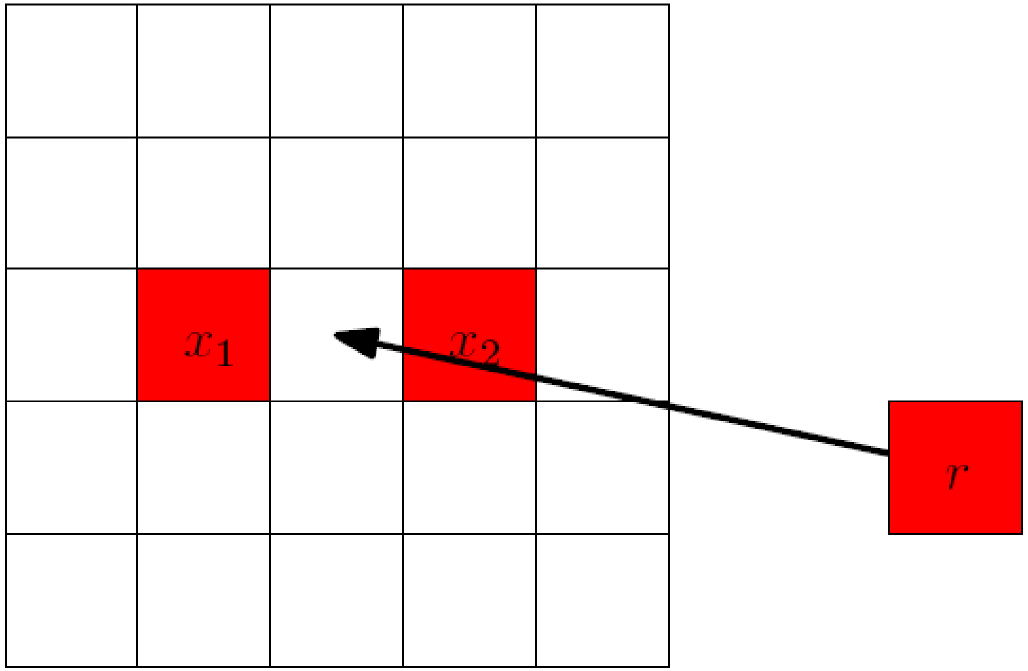





RGB Image
Mosaic Raw Image
Demosaicing

- Demosaicing: estimate the missing colour channels
- Use the neighbourhood pixels values to make prediction
- Example: estimate the missing blue and red channels for pixel at the centre of the given mosaic patch





- Prior: natural images are smooth
- Model: $\tilde{r} = a_1 x_1 + a_2 x_2$
- Data: 3 adjacent red pixels (X_{i1}, r_i, X_{i2}) for $i \in [1, n]$
- Prediction Error (Loss Function):

$$\mathcal{L} = \sum_{i=1}^{n} (\tilde{r}_i - r_i)^2 = \sum_{i=1}^{n} (a_1 X_{i1} + a_2 X_{i2} - r_i)^2$$

- Minimize the prediction error

$$\min_{a_1,a_2} \sum_{i=1}^{n} (a_1 X_{i1} + a_2 X_{i2} - r_i)^2 = \|XA - R\|_2^2$$

where

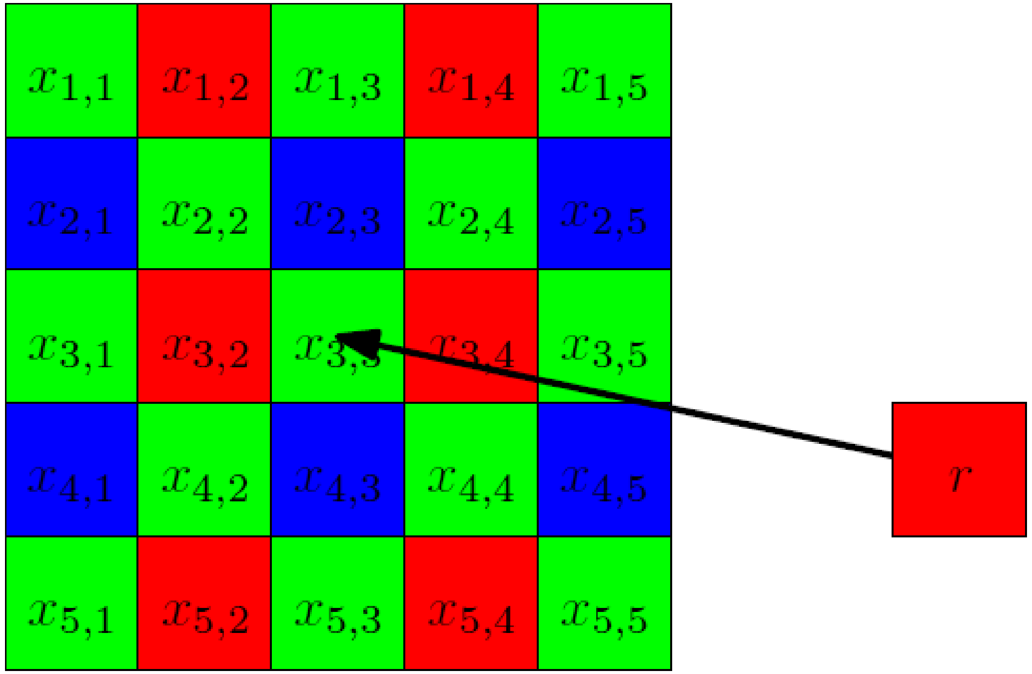
$$X = \begin{bmatrix} X_{11} & X_{12} \\ X_{21} & X_{22} \\ \vdots & \vdots \\ X_{n1} & X_{n2} \end{bmatrix}, \quad A = \begin{bmatrix} a_1 \\ a_2 \end{bmatrix}, \quad R = \begin{bmatrix} r_1 \\ r_2 \\ \vdots \\ r_n \end{bmatrix},$$

- Find extrema

$$\frac{\partial \|XA - R\|_2^2}{\partial A} = 0 \quad \Longrightarrow \quad X^\top (XA - R) = 0$$

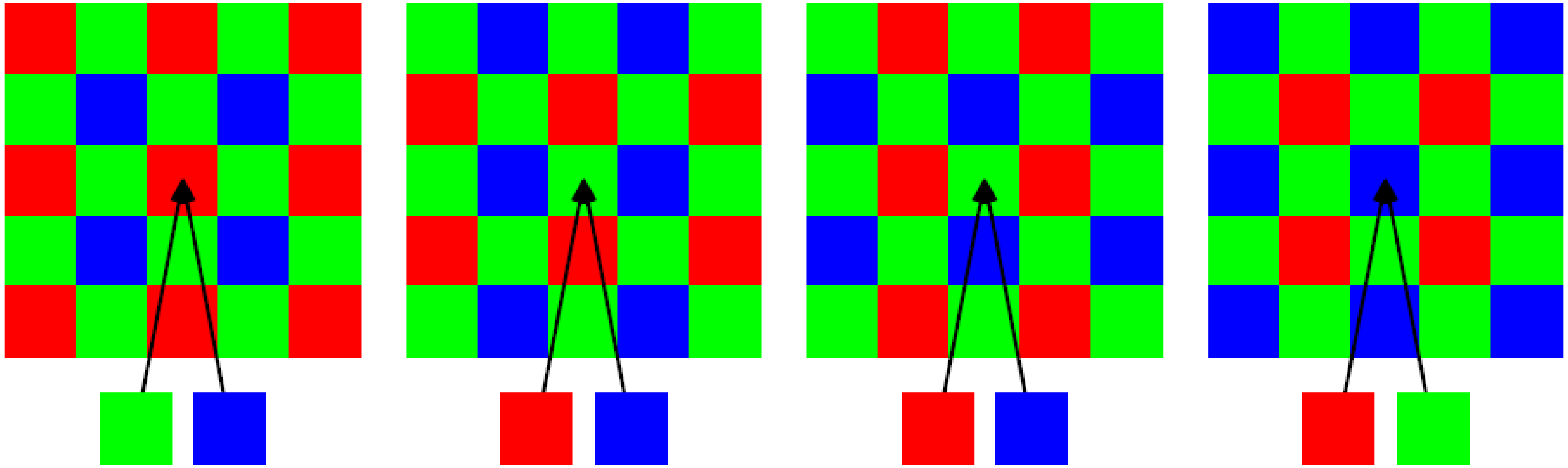
- Solution

$$A = (X^\top X)^{-1} X^\top R = X^+ R$$



- Prior:
  1. Natural images are smooth
  2. Colour channels are highly correlated
- Model: $\tilde{r} = \sum_{i=1}^{5} \sum_{j=1}^{5} a_{ij} x_{ij}$
- Reshape patch to column vector
- Solution: $A = X^{+} R$

- 4 types of mosaic patches, each with 2 missing colour channels



- 8 coefficient matrices are required in total
- Helpful functions:
  - `im2col(...)`, rearrange image blocks into columns
  - `colfilt(...)`, columnwise neighborhood operations
  - `imfilter(...)`, image filtering

Rearrange 2D 5×5 image into 1D vector

$$\begin{pmatrix} x_{1,1,1} & x_{1,1,2} & x_{1,1,3} & x_{1,1,4} & x_{1,1,5} \\ x_{1,2,1} & x_{1,2,2} & x_{1,2,3} & x_{1,2,4} & x_{1,2,5} \\ x_{1,3,1} & x_{1,3,2} & x_{1,3,3} & x_{1,3,4} & x_{1,3,5} \\ x_{1,4,1} & x_{1,4,2} & x_{1,4,3} & x_{1,4,4} & x_{1,4,5} \\ x_{1,5,1} & x_{1,5,2} & x_{1,5,3} & x_{1,5,4} & x_{1,5,5} \end{pmatrix} \rightarrow \begin{pmatrix} x_{1,1,1} \\ x_{1,1,2} \\ x_{1,1,3} \\ x_{1,1,4} \\ x_{1,1,5} \\ x_{1,2,1} \\ x_{1,2,2} \\ x_{1,2,3} \\ \vdots \\ x_{1,5,3} \\ x_{1,5,4} \\ x_{1,5,5} \end{pmatrix}$$

1st index is pixel patch number

Convert it into a row vector

$$\begin{pmatrix} x_{1,1,1} \\ x_{1,1,2} \\ x_{1,1,3} \\ x_{1,1,4} \\ x_{1,1,5} \\ x_{1,2,1} \\ x_{1,2,2} \\ x_{1,2,3} \\ \vdots \\ x_{1,5,3} \\ x_{1,5,4} \\ x_{1,5,5} \end{pmatrix} \rightarrow \left(x_{1,1} x_{1,2} x_{1,3} x_{1,4} x_{1,5} \cdots x_{1,23} x_{1,24} x_{1,25} \right)$$

Turn 2D indices to single index

Stack many rows together into a huge matrix

$$\begin{pmatrix} x_{1,1} x_{1,2} x_{1,3} \cdots x_{1,25} \\ x_{2,1} x_{2,2} x_{2,3} \cdots x_{2,25} \\ x_{3,1} x_{3,2} x_{3,3} \cdots x_{3,25} \\ \vdots \qquad \ddots \quad \vdots \\ x_{n,1} x_{n,2} x_{n,3} \cdots x_{n,25} \end{pmatrix}$$

n is the number of mosaic patches for training.

Linear regression

$$\begin{pmatrix} x_{1,1} x_{1,2} x_{1,3} \cdots x_{1,25} \\ x_{2,1} x_{2,2} x_{2,3} \cdots x_{2,25} \\ x_{3,1} x_{3,2} x_{3,3} \cdots x_{3,25} \\ \vdots \qquad \ddots \quad \vdots \\ x_{n,1} x_{n,2} x_{n,3} \cdots x_{n,25} \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \\ a_3 \\ \vdots \\ a_{25} \end{pmatrix} = \begin{pmatrix} g_1 \\ g_2 \\ g_3 \\ \vdots \\ g_n \end{pmatrix}$$

$$i.e., \boldsymbol{X}\boldsymbol{a} = \boldsymbol{g},$$

$$\boldsymbol{a} = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{g}$$

Linear regression

thousands of 5x5 image patches of the same mosaic type

the first 5x5 image patch

demosaicing coefficient

the missing green component

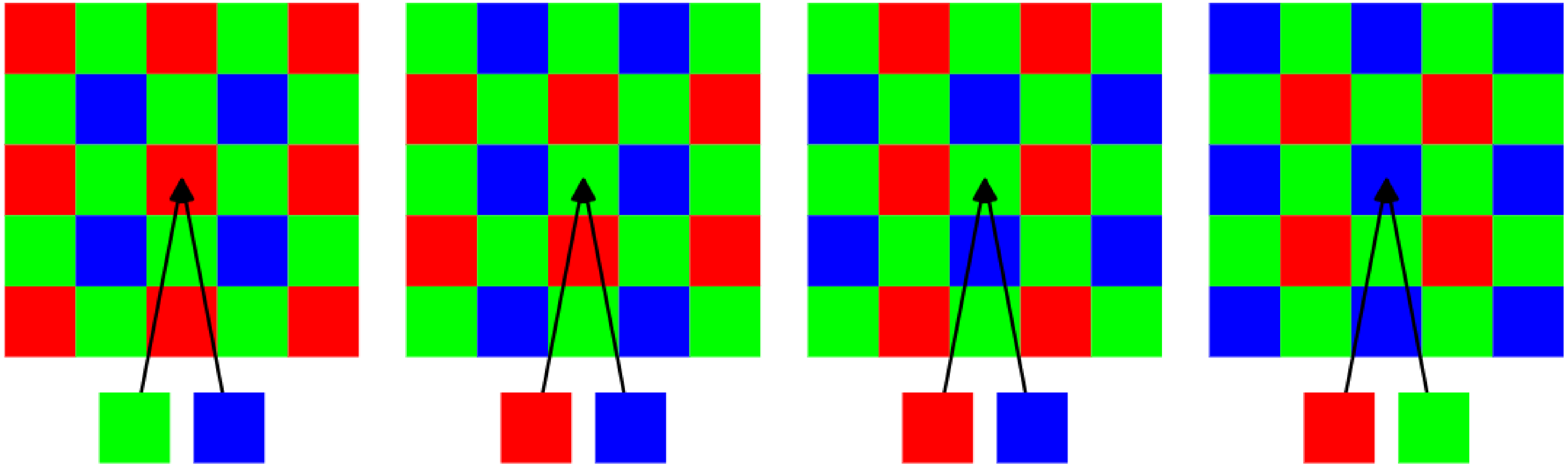
$$
\begin{pmatrix}
\boxed{x_{1,1}\, x_{1,2}\, x_{1,3}\, \cdots\, x_{1,25}} \\
x_{2,1}\, x_{2,2}\, x_{2,3}\, \cdots\, x_{2,25} \\
x_{3,1}\, x_{3,2}\, x_{3,3}\, \cdots\, x_{3,25} \\
\vdots \qquad \ddots \quad \vdots \\
x_{n,1}\, x_{n,2}\, x_{n,3}\, \cdots\, x_{n,25}
\end{pmatrix}
\boxed{\begin{pmatrix} a_1 \\ a_2 \\ a_3 \\ \vdots \\ a_{25} \end{pmatrix}}
=
\begin{pmatrix} \boxed{g_1} \\ g_2 \\ g_3 \\ \vdots \\ g_{25} \end{pmatrix}
$$

$$i.e., \boldsymbol{X}\boldsymbol{a} = \boldsymbol{g},$$

$$\boldsymbol{a} = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{g}$$

Fitting eight times for eight different demosaicing types

4 types of mosaic patches, each with 2 missing colour channels



8 coefficient matrices are required in total

Regression for machine learning

- In the training stage, use a large number of true RGB images (ground truth) to learn the statistical model; namely, the linear regression function.
- In the inference stage, use the trained regression model to predict or recover the missing color samples.