

Working with Multidimensional Data Using NumPy

EXPLORING MULTIDIMENSIONAL DATA USING NUMPY



Janani Ravi

CO-FOUNDER, LOONYCORN

www.loonycorn.com

Overview

Fundamental package for scientific computing in Python

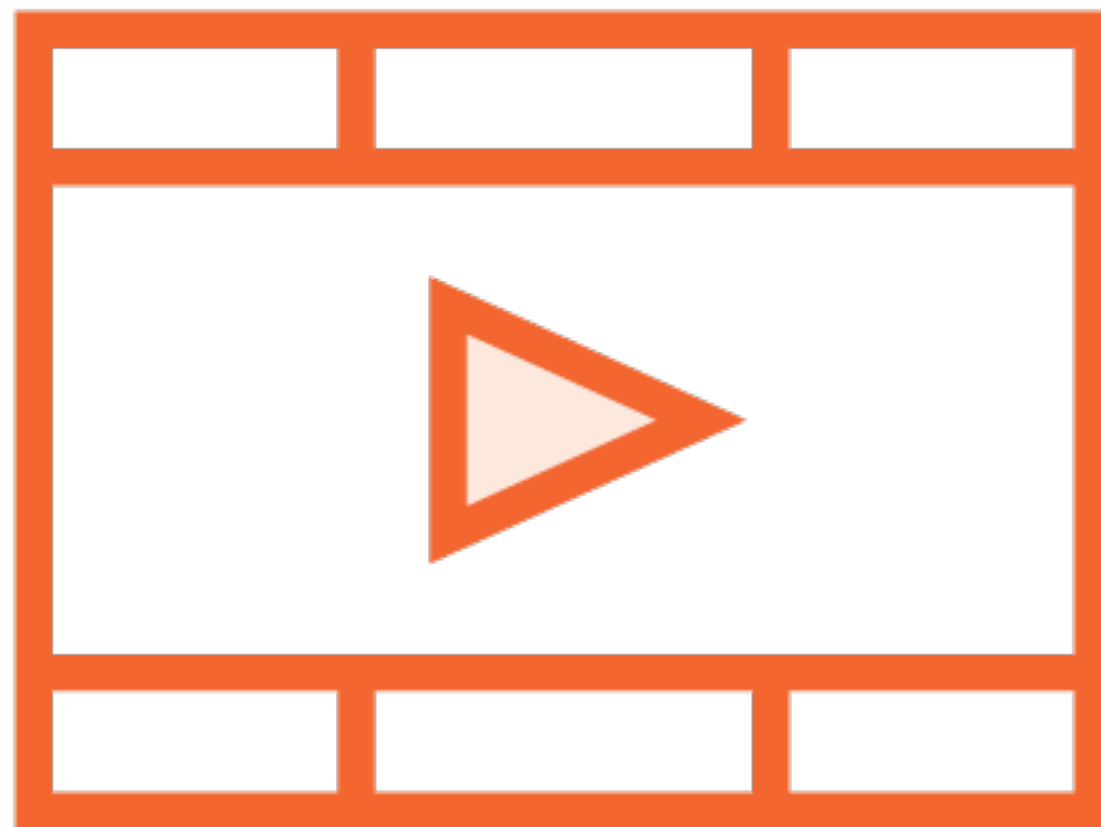
Basic building block is a powerful n-dimensional array

Forms the core of the Python ecosystem of open source software for math, science and engineering

Offers easy to use functions to process multi-dimensional arrays

Prerequisites and Course Outline

Prerequisite Courses



Python: Getting Started

Python Fundamentals

Advanced Python



Software and Skills

Be very comfortable programming in Python (Python 3)

Be comfortable working with Jupyter notebooks

Understand high school matrix operations



Course Outline

Exploring multidimensional data

- Creating, printing, basic operations
- Shape manipulation, deep and shallow copies

Complex indexing

- Indexing using array indices
- Broadcasting, stacking vectors
- Support Vector Machines for text and image classification, Gradient Boosting for regression

NumPy with other libraries

- SciPy and Pandas
- KNN with TensorFlow

NumPy Ecosystem

statsmodel

Estimate statistical models,
and perform tests

scikit-image

Collection of algorithms for
image processing

scikit-learn

Simple and efficient tools for
machine learning in Python

pandas

Data analysis and
manipulation

matplotlib

Plotting library for 2D graphs
and visualizations

NumPy Ecosystem

statsmodel

Estimate
statistical
models, and
perform tests

scikit-image

Collection of
algorithms for
image
processing

scikit-learn

Simple and
efficient tools
for machine
learning in
Python

pandas

Data analysis
and
manipulation

matplotlib

Plotting library
for 2D graphs
and
visualizations

NumPy

Demo

Creating multidimensional arrays

Demo

Printing arrays

Demo

Basic arrays operations

Demo

**Universal functions to perform familiar
mathematical operations**

Demo

Indexing and slicing of arrays

Demo

Iterating over arrays

Demo

Changing the shape of an array

Demo

Splitting arrays

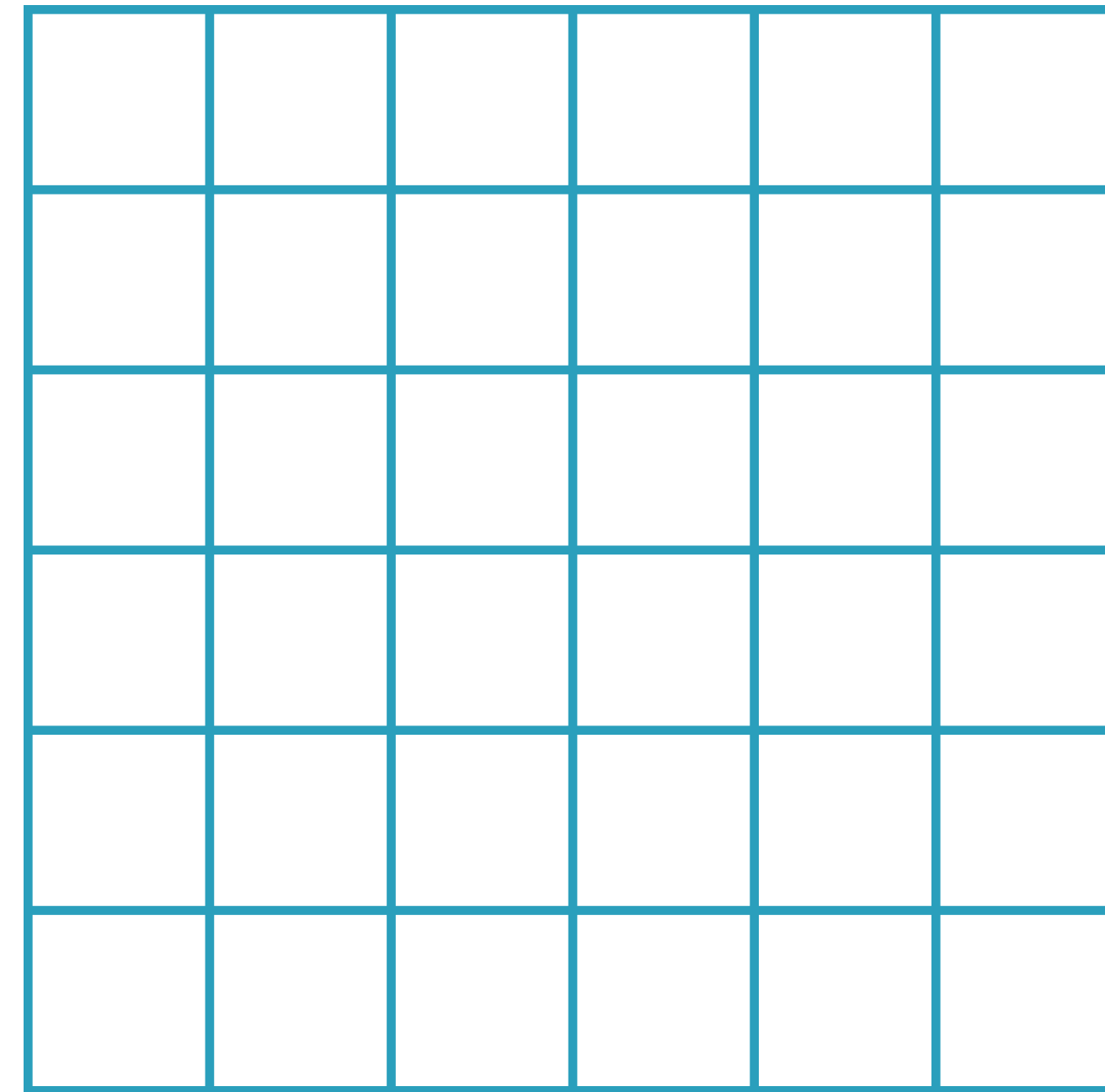
Demo

Image manipulation

Images as Arrays



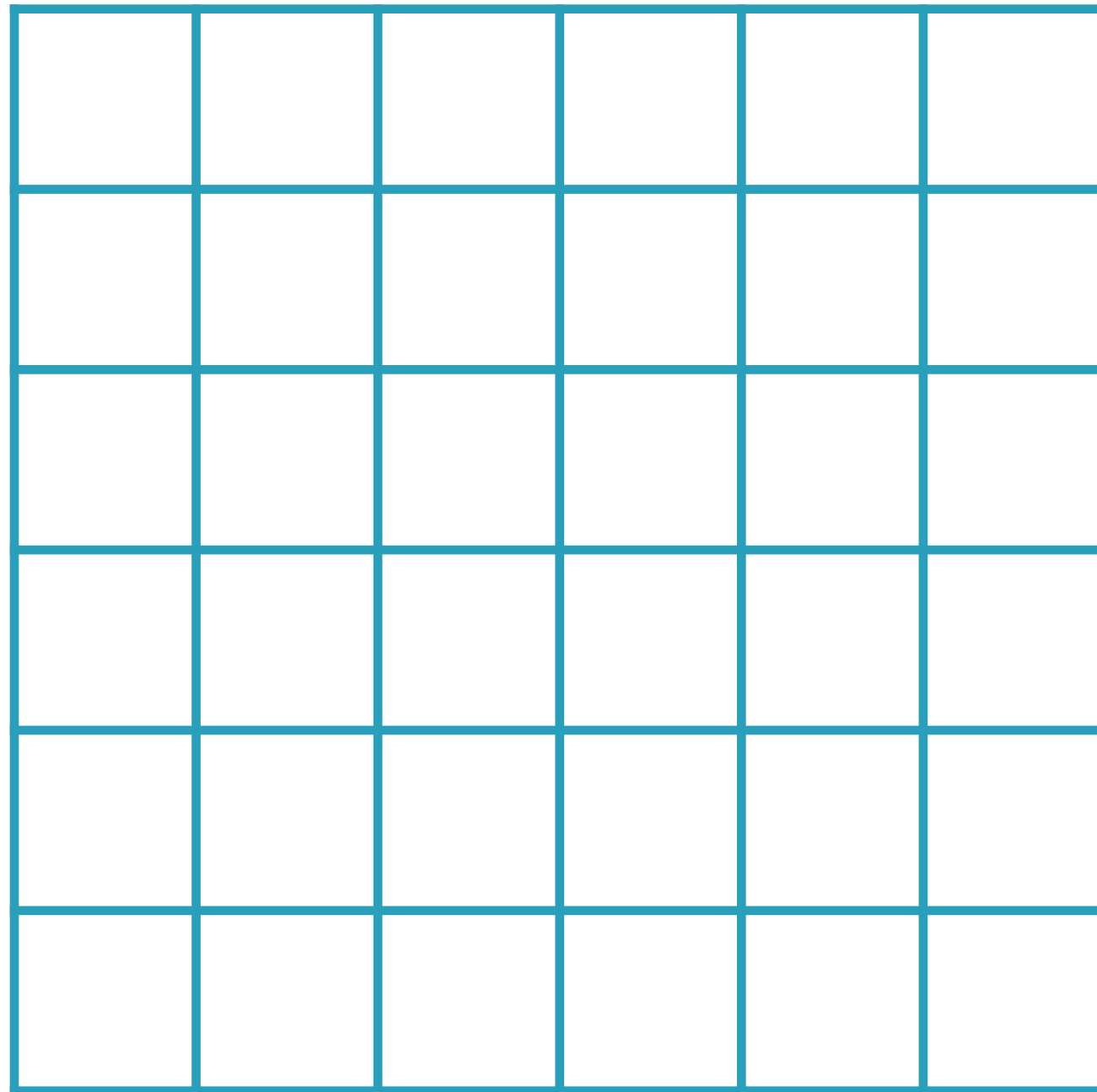
Images as Arrays



Each pixel holds a value based on the type of image



RGB Images

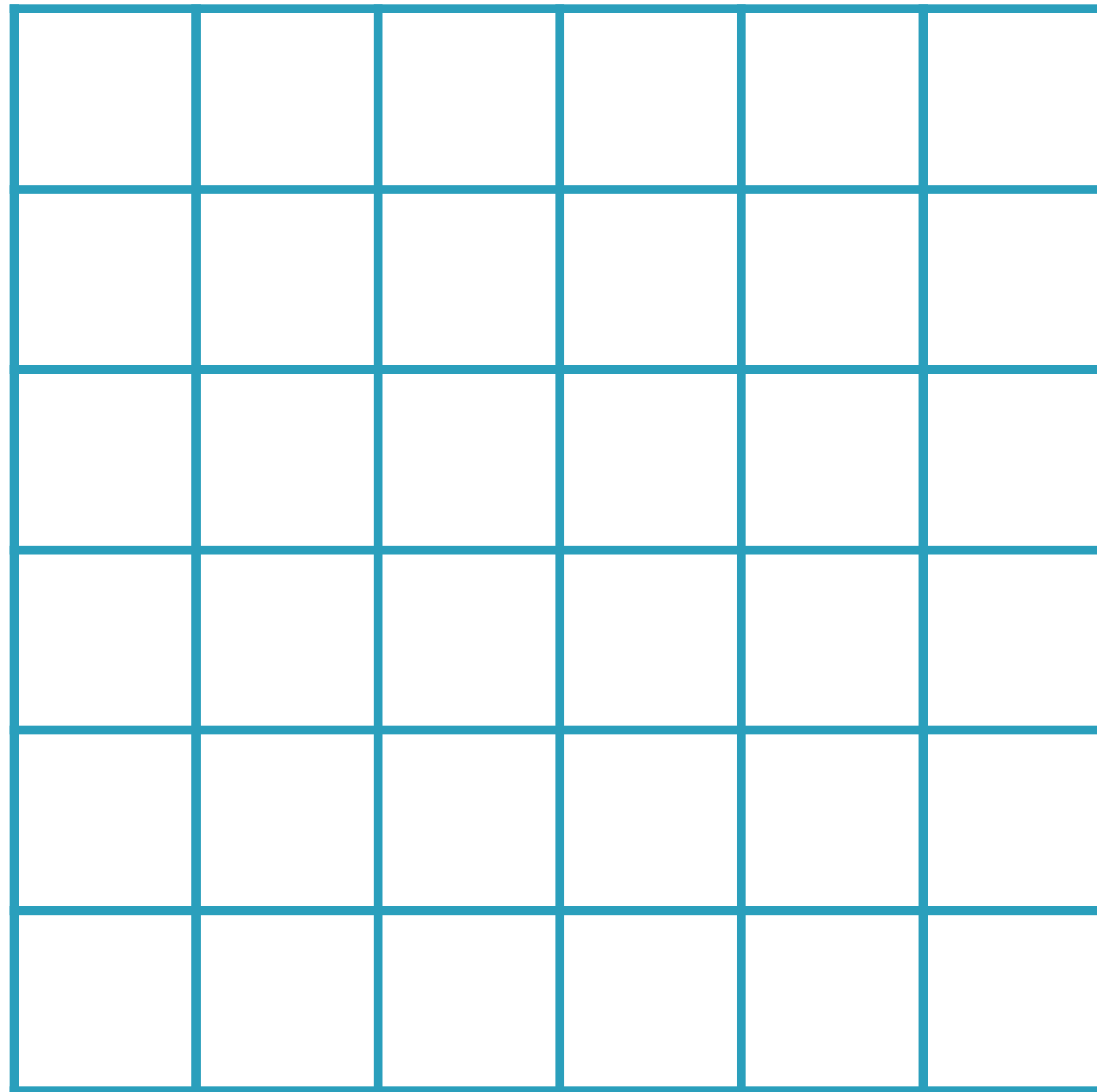


**RGB values are
for color images**

R, G, B: 0-255



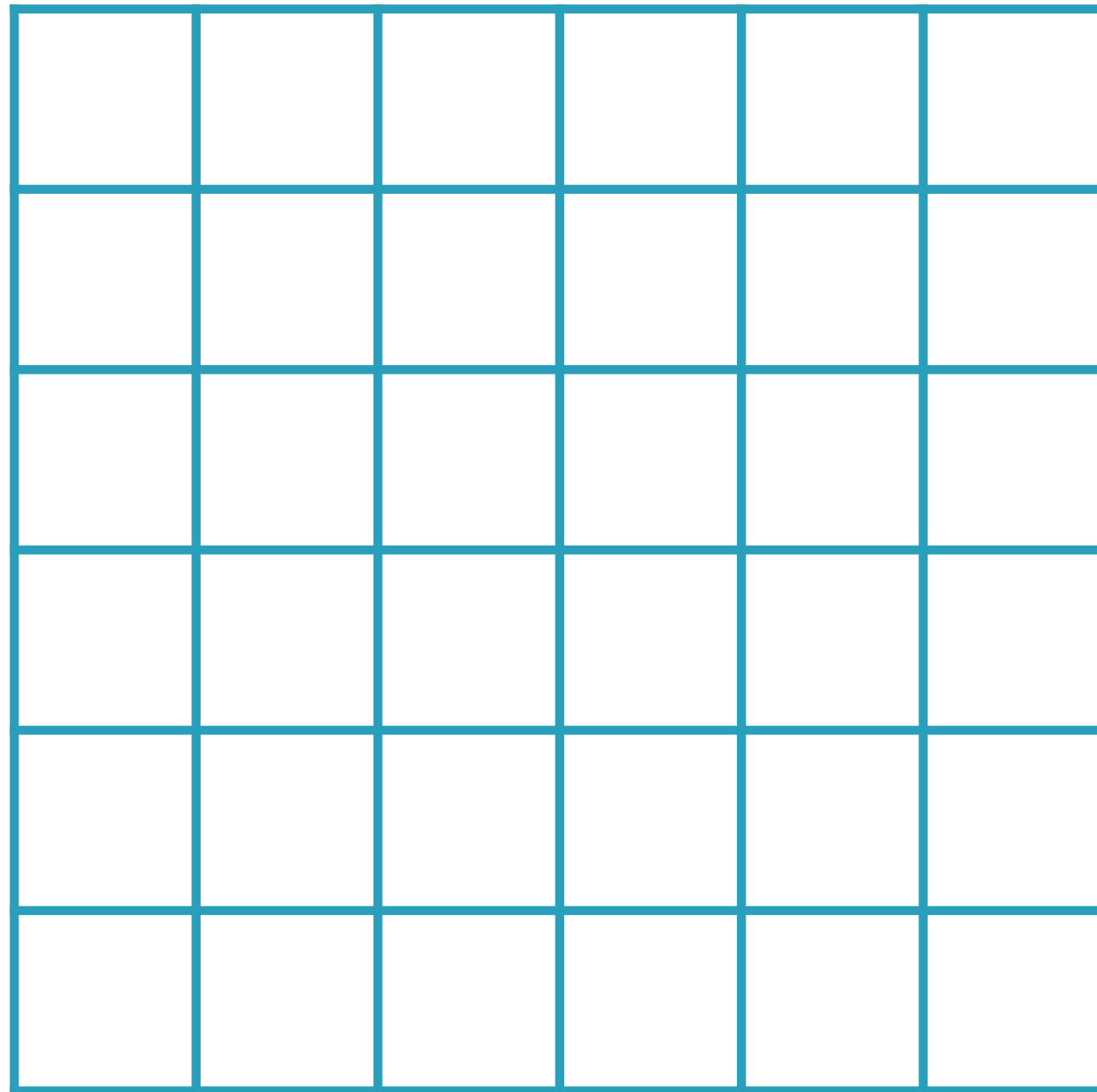
RGB Images



255, 0, 0



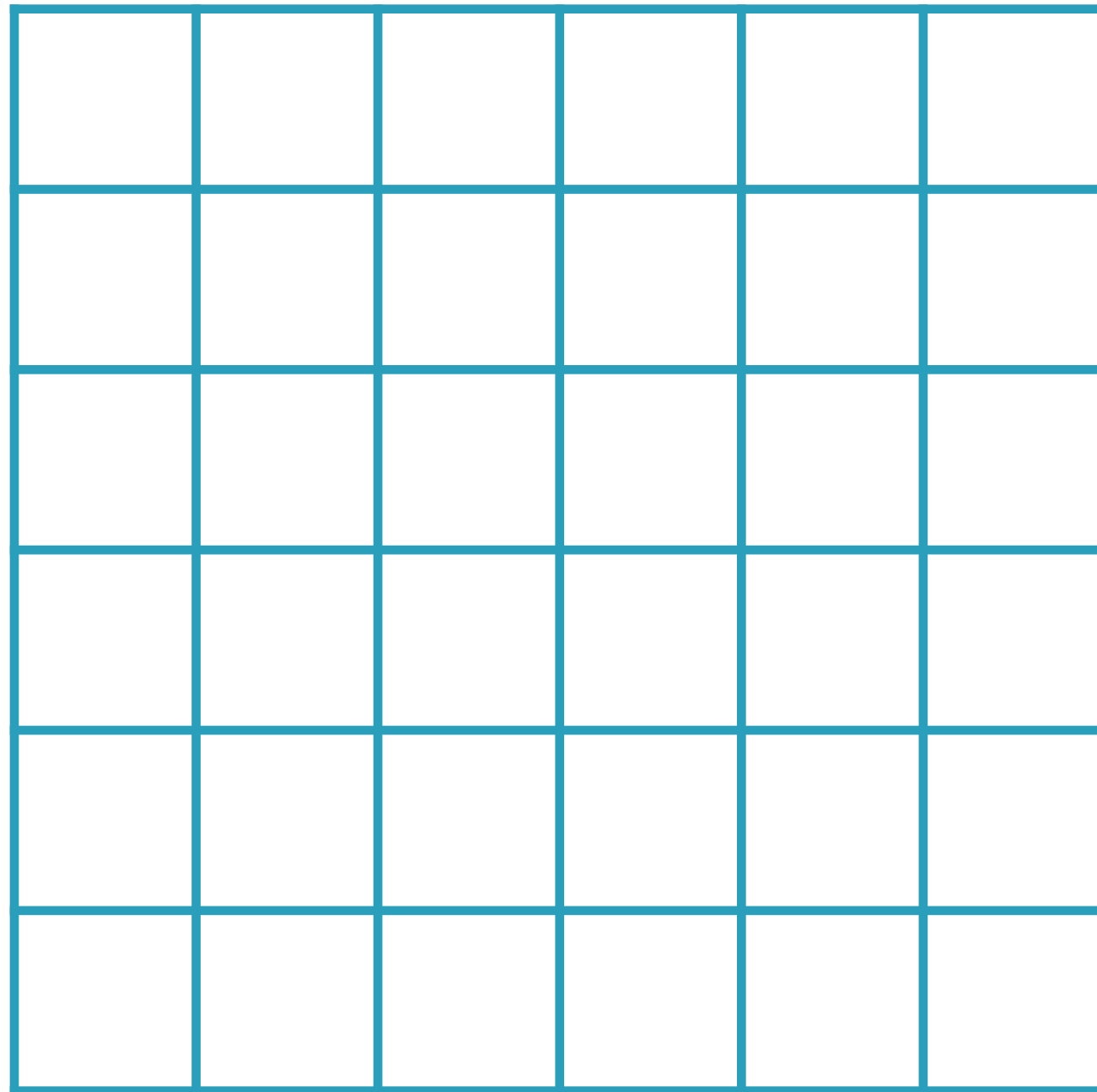
RGB Images



0, 255, 0



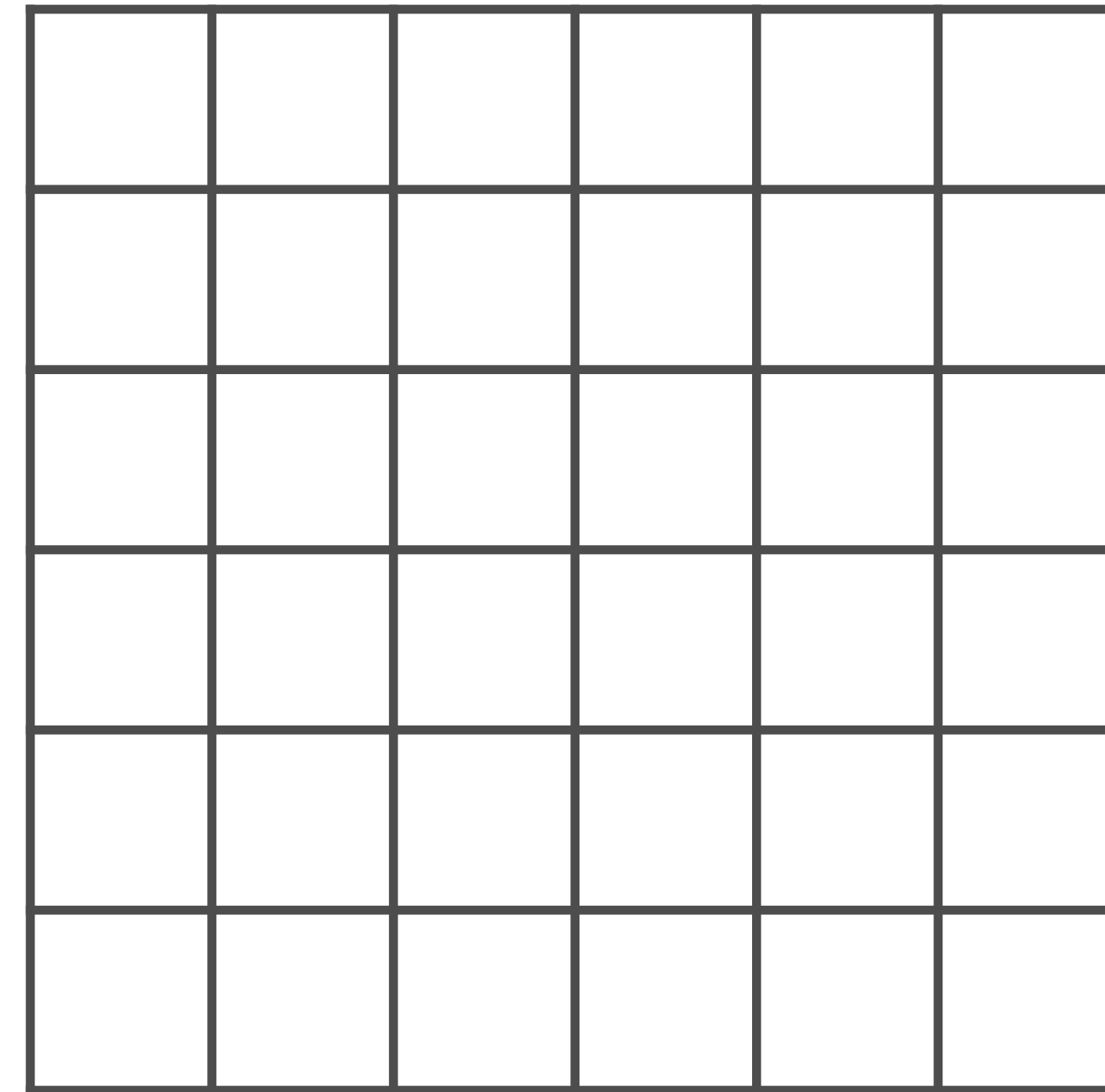
RGB Images



0, 0, 255

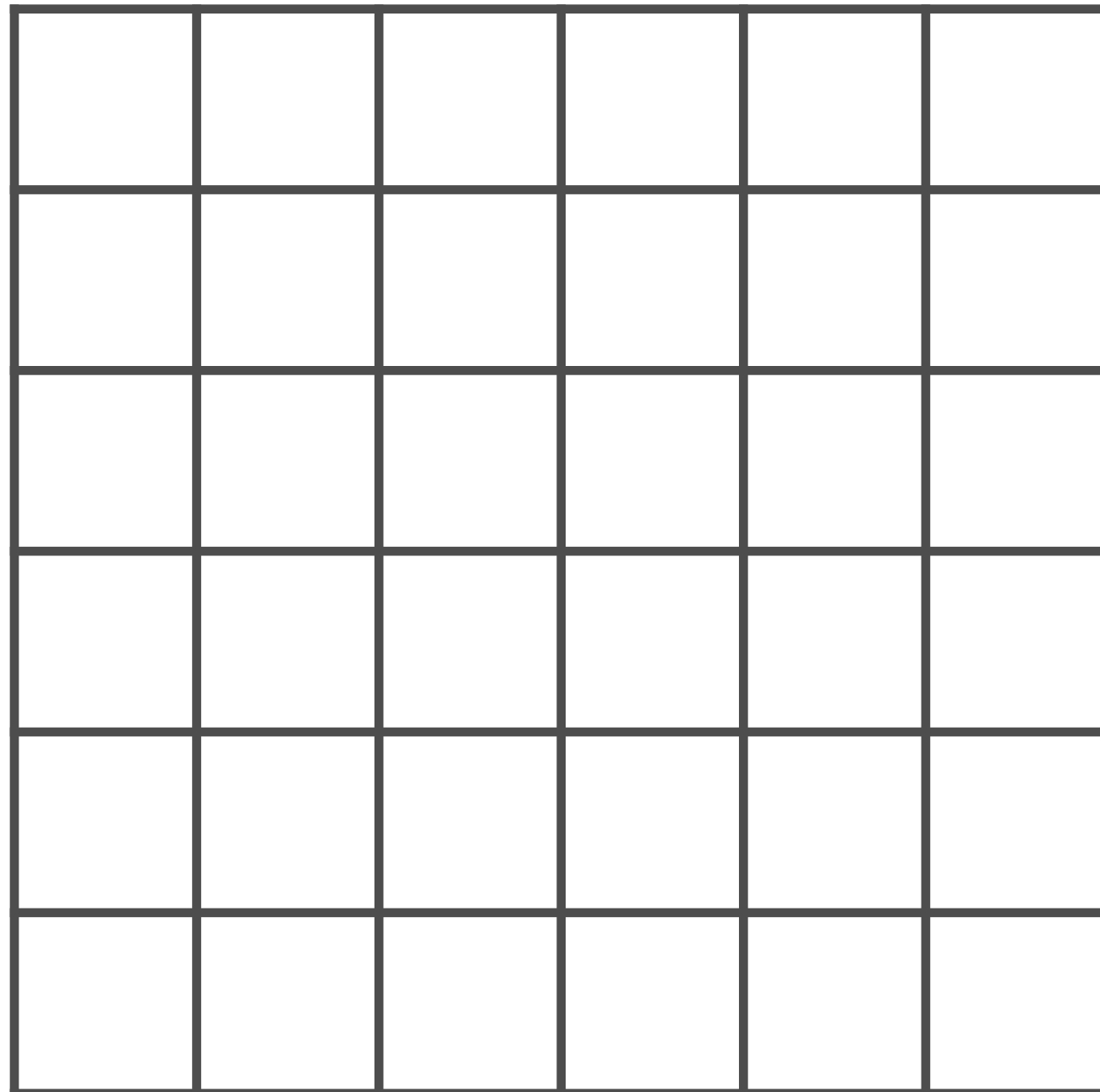
3 values to represent
color, **3** channels

Grayscale Images





Grayscale Images

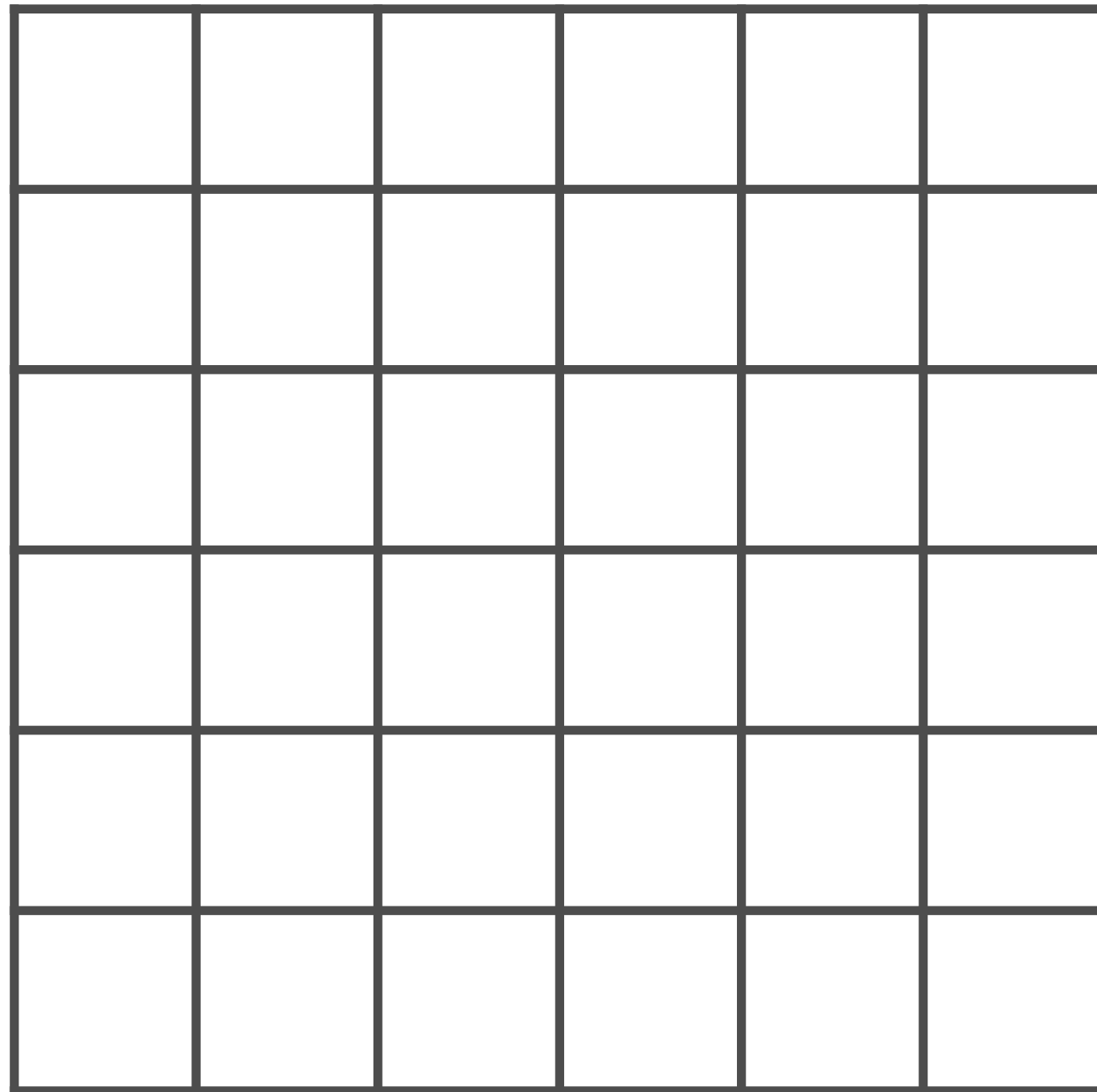


**Each pixel represents
only intensity information**

0.0 - 1.0

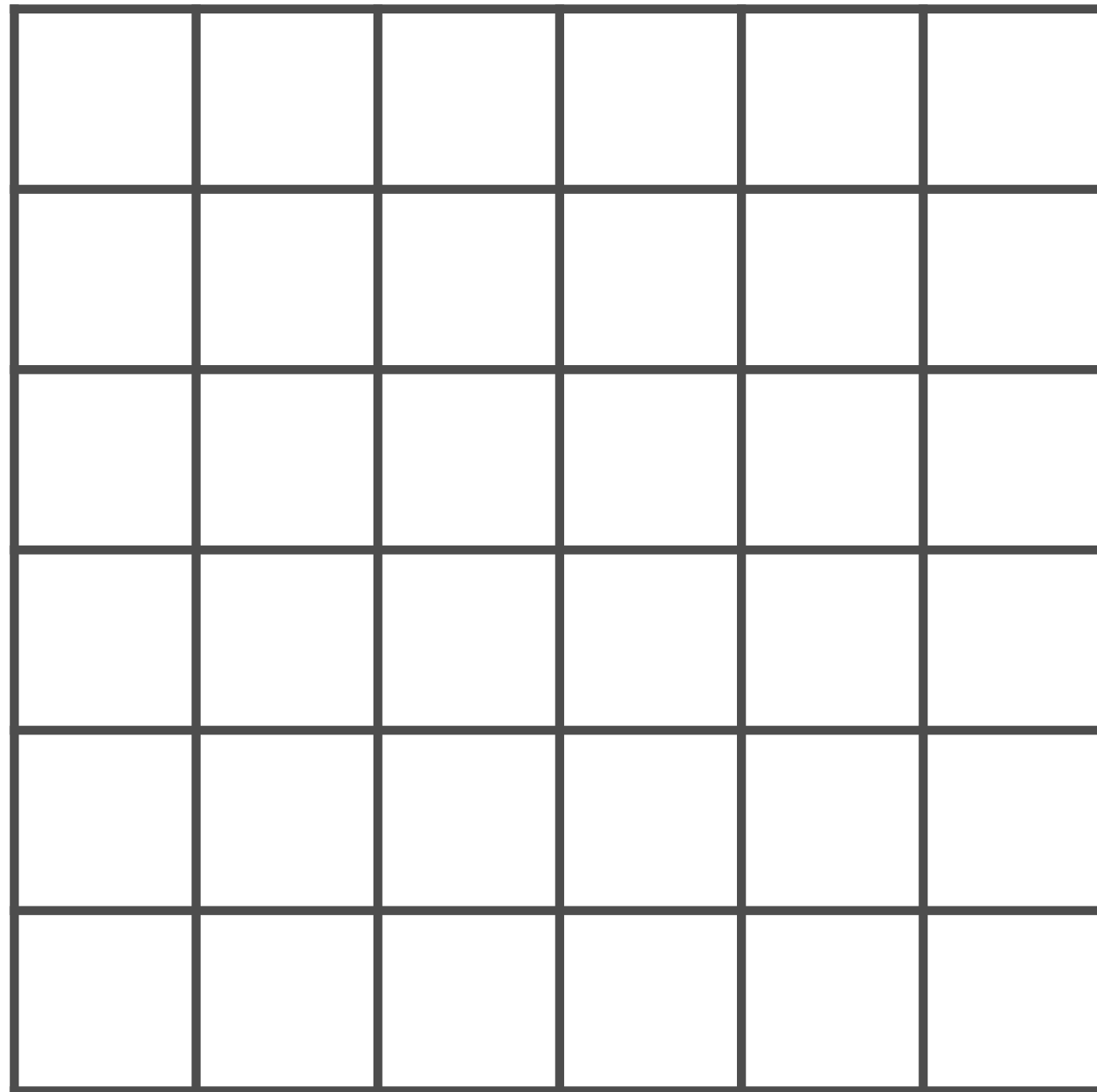


Grayscale Images





Grayscale Images



0.5

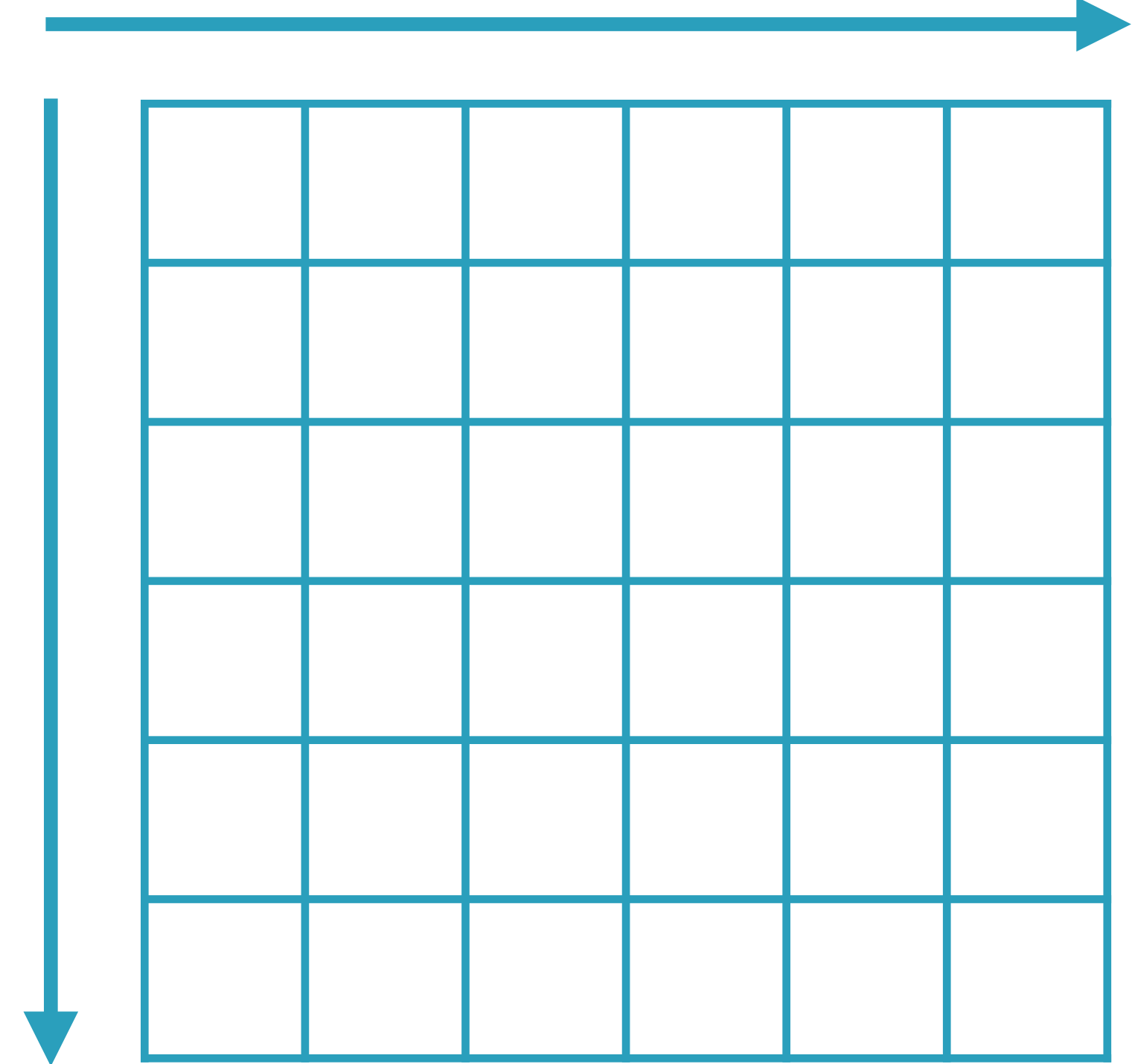
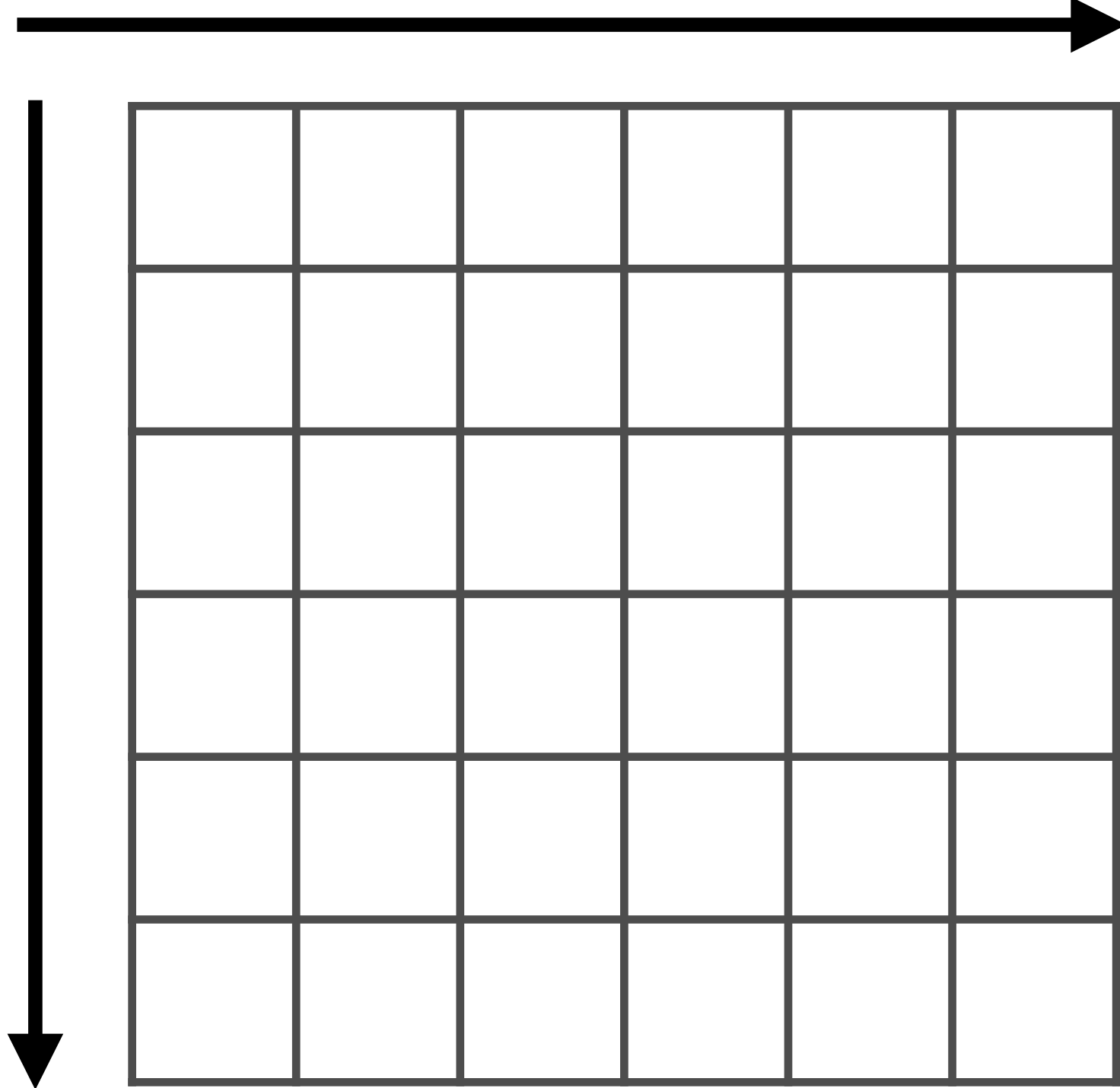
1 value to represent
intensity, **1** channel

Images as Arrays



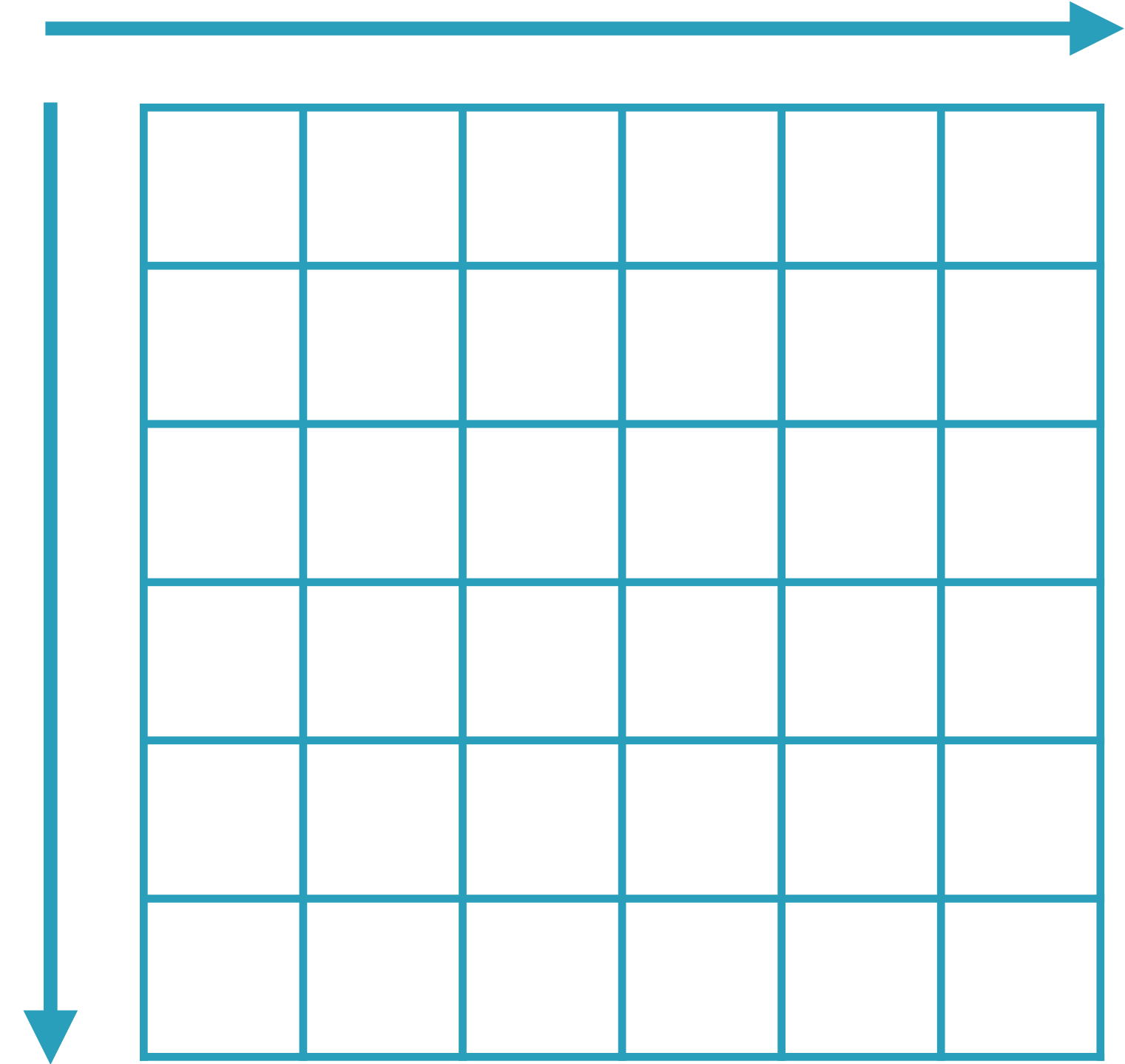
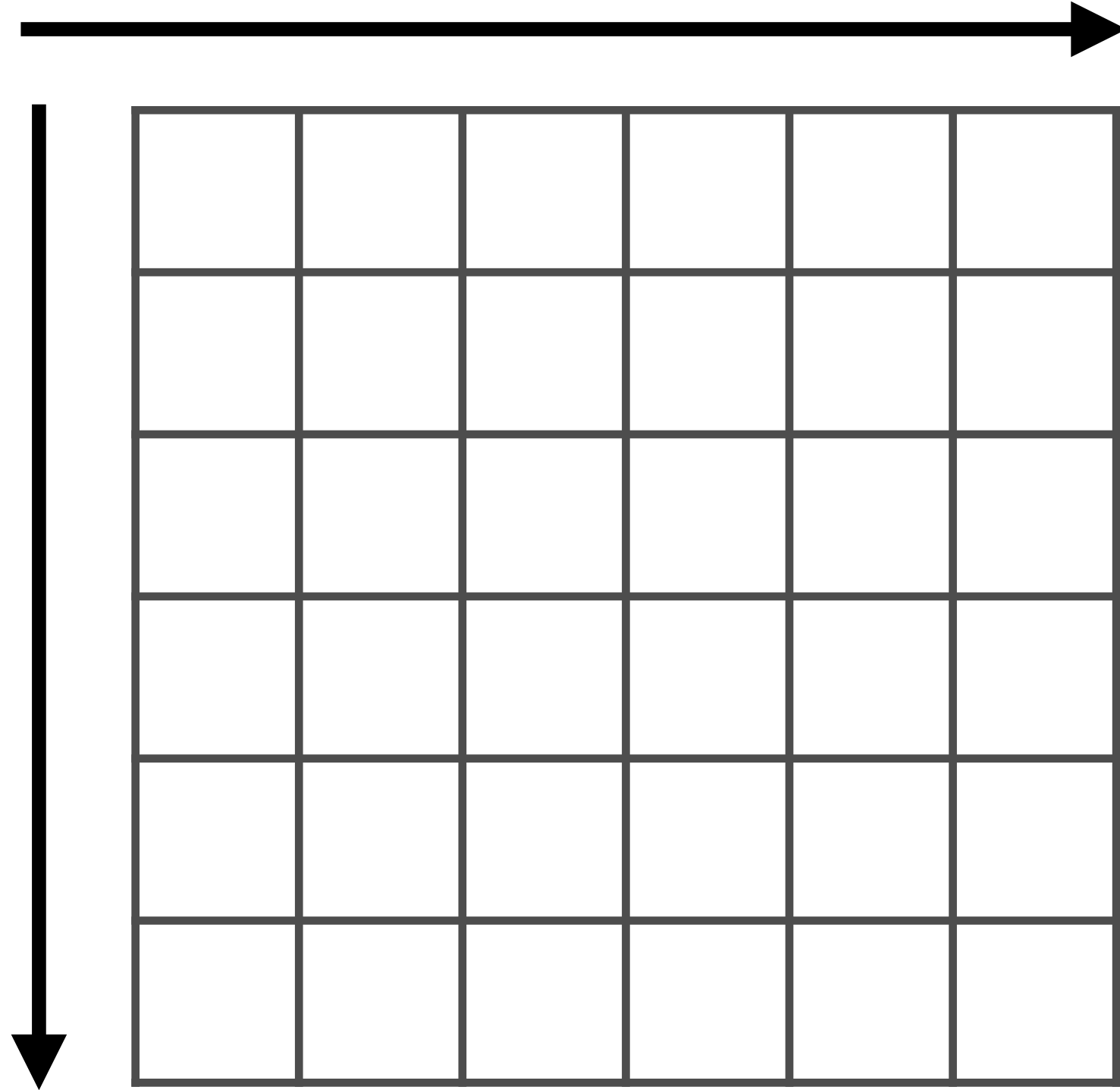
Single channel and multi-channel images

Images as Arrays



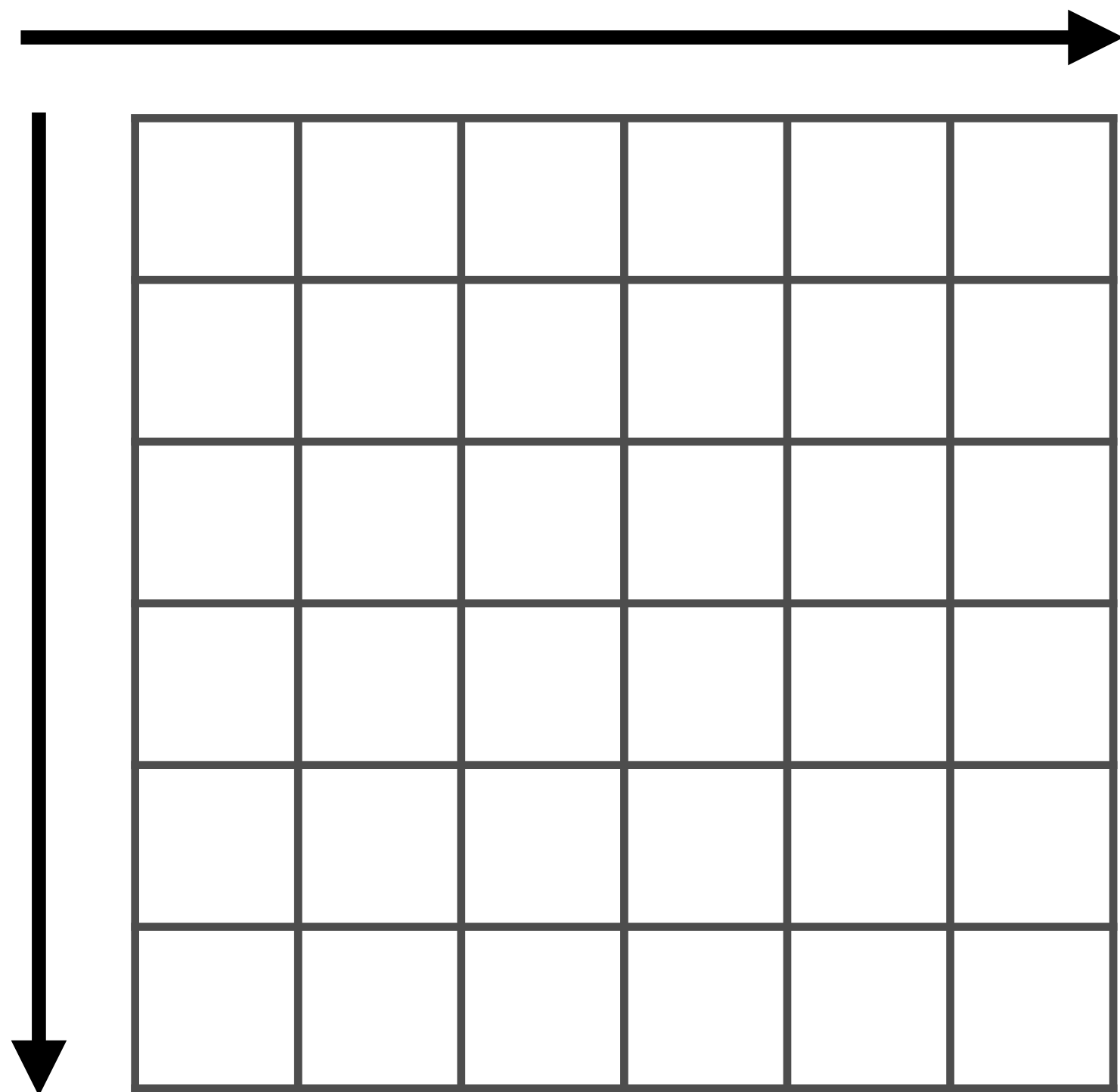
Images can be represented by a 3-D matrix

Images as Arrays

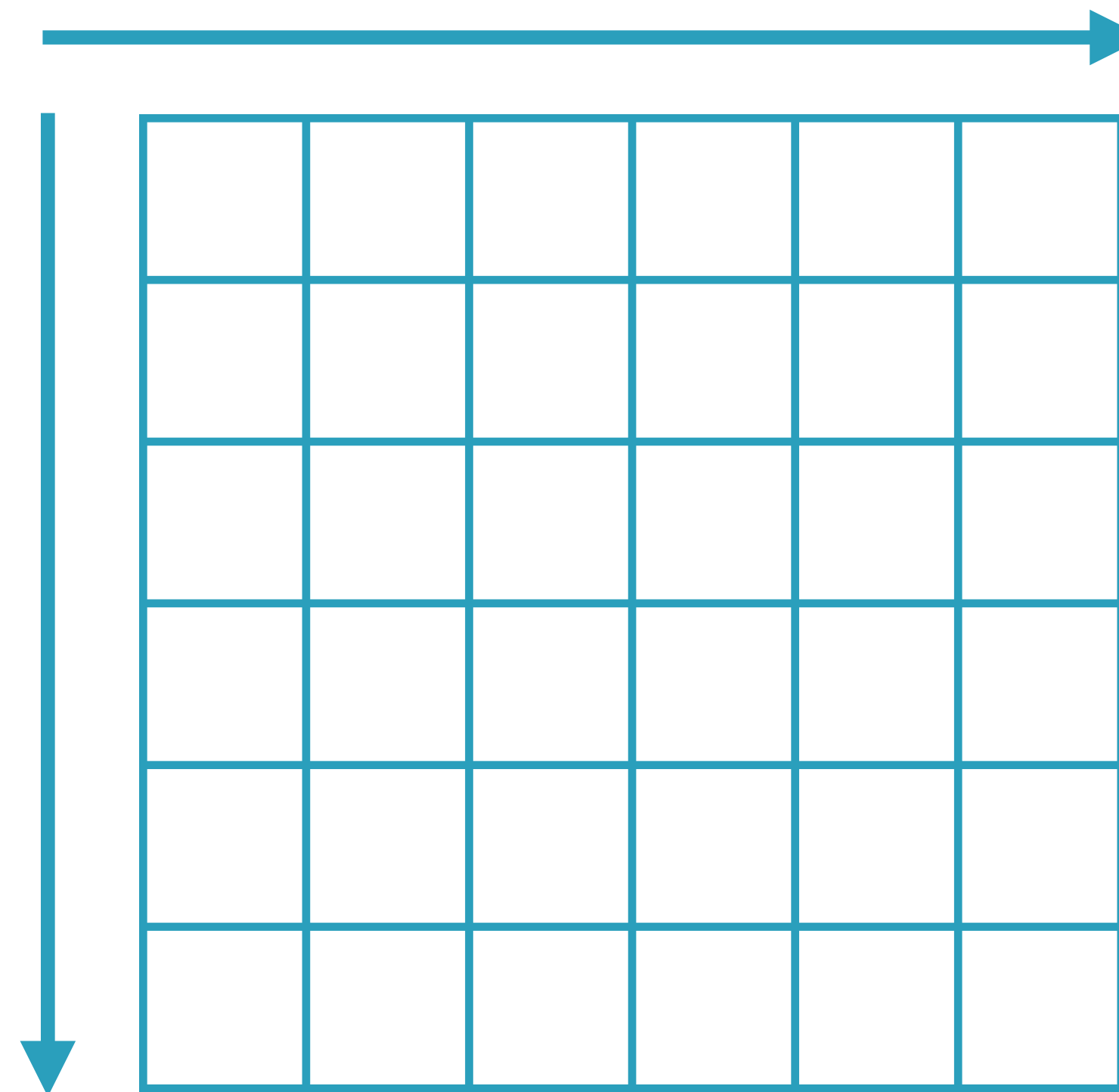


The **number of channels** specifies the **number of elements** in the 3rd dimension

Images as Arrays



(6, 6, 1)



(6, 6, 3)

Demo

Views - shallow copies of an array

Demo

Making deep copies of arrays

Summary

Fundamental package for scientific computing in Python

Basic building block is a powerful n-dimensional array

Offers easy to use functions to process multi-dimensional arrays

Basic operations, universal functions, reshaping, splitting, making shallow and deep copies