



Monokai Classic Marp theme

Monokai Classic

A dark theme for [Marp](#) inspired by the Monokai Classic theme.

Adapted from the Dracula theme by
Daniel Nicolas Gisolfi.

brayevalerien

Equations

There is support for inline equations which allow to define $z \hookrightarrow \mathcal{N}(0, \mathbf{I})$, and use it in a follow up display equation like the following.

$$\begin{aligned}x_t &= \sqrt{\bar{\alpha}_t} x_0 + \sqrt{1 - \bar{\alpha}_t} z \\x_{t-1} &= \frac{1}{\sqrt{\alpha_t}} \left(x_t - \frac{1 - \alpha_t}{\sqrt{1 - \bar{\alpha}_t}} z \right) + \sigma_t \epsilon\end{aligned}$$

Note that this document uses `mathjax` but `katex` can be used too, changing the properties at the top of the source code.

Tables

Tables are inspired by the clean *LATEX* tables.

Country	Industry	Tool	Daily active users
USA	Technology	ChatGPT	2461
France	Manufacturing	Midjourney	8496
UK	Agriculture	Stable Diffusion	1124
Brazil	Agriculture	Midjourney	3267a
USA	Healthcare	Bard	4265
China	Retail	Midjourney	613

[source](#)

Figures

Images are resized to fit the page, but it still isn't optimal. Add more lines here to see how the image can go out of the slide.



Multiple figures

Use html and css as a hacky way to get multiple figures next to each others.



This isn't the best solution, if you have anything better please [open an issue](#).

Blockquotes

As a wise man (Einstein) once said,

Pure mathematics is, in its way, the poetry of logical ideas.

Multi-line quotes are supported too:

I wonder where I'll float next?

— The barrel boy

Lists

Unordered

Unordered lists with nesting are supported.

- CNNs
 - Convolution layers
 - Edge detection filters
 - Feature maps
 - Pooling layers
 - Max pooling
 - Average pooling
- Data augmentation
 - Rotation
 - Flipping

Ordered

Ordered lists with nesting are supported too.

1. Preprocessing

1. Normalization
2. Augmentation

2. Feature Extraction

1. Edge Detection
2. Texture Analysis

3. Models

1. CNN
 1. Conv Layers
 2. Pooling Layers
2. Transfer Learning

Code formating

Surround text with backticks like `theme.set("monokai")` or use tripple backticks.

```
import torch.nn as nn
import torch

conv = nn.Conv2d(in_channels=3, out_channels=16, kernel_size=3, stride=1, padding=1)
x = torch.randn(1, 3, 64, 64)
out = conv(x)
print(out.shape) # Expect torch.Size([1, 16, 64, 64])
```

Language can be specified after the first triple backticks.