

# King of Dumbs

笨蛋之王

级别: ♣ 新手

## 电脑怎样把照片变成数字？

一张照片就是几千万个数，人工智能对它做的几乎全是乘法和加法。分三个层次讲向量、点积和矩阵，配两个互动小工具。

中文 发布于 2026年10月17日



扫码打开可互动的课程

<https://kingofdumbs.com/zh/lessons/vectors-and-matrices>

## 你将学到

- 照片就是一列数：向量和维数
- 点积：乘一乘再加起来，以及两支箭头有多同向
- 矩阵会挪动平面：它的列、面积和剩下的维数
- 矩阵乘法：把变换串起来，为什么顺序要紧，以及神经网络在做什么

### 本课会出现的符号

$E_a$  下标：右下角的标签 (♠) ·  $\Sigma$  求和：全部加起来 (♠) ·  $x_i$  下标  $i$ ：第  $i$  个元素 (♠) ·  $\approx$  约等于 (♠) ·  $\leq \geq$  小于等于 / 大于等于 (♠) ·  $|x|$  绝对值：去掉正负号 (♠) ·  $\cdot$  点：乘（或点积） (♠♠) ·  $\sigma$  sigma：含义因课而异 (♠) ·  $\theta$  theta：通常是角度 (♠) ·  $\lambda$  lambda：一个缩放用的数 (♠) ·  $\|x\|$  范数：箭头的长度 (♠) ·  $x^T$  转置：把列放倒 (♠) ·  $\in$  属于 (♠) ·  $\mathbb{R} \mathbb{Z}$  数集 (♠)

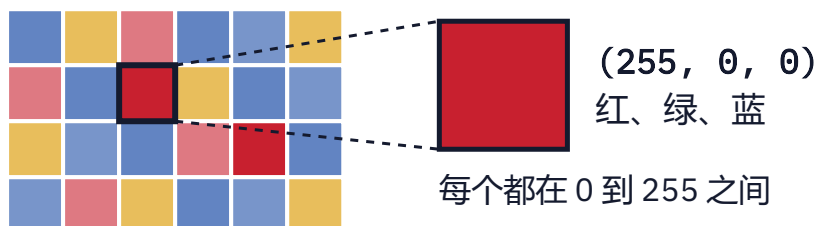
你给猫拍了张照片，在手机相册里搜”猫”，很多时候就能把它找出来。可电脑并看不见猫：它唯一会做的，就是飞快地加数和乘数。

这一课的傻问题是：**电脑怎样把照片变成数字？** 顺便再问一句：变成数字以后，它拿这些数做什么？答案就是三个听起来像考试题的概念：向量、点积和矩阵；再加上一道算术：乘一乘，再加起来。

选一张牌：♣ 从零开始，♦ 不怕简单的公式，♠ 想看符号。随时可以换。



## 照片就是一列数



**$4000 \times 3000 \times 3 = 3600$  万个数**  
排成一行：一个 3600 万维的向量

数码照片就像一幅马赛克：它由一个个彩色小方块拼成，这些小方块叫**像素**，小到肉眼看不出来。电脑打开照片时，会把每个小方块存成三个数：红色有多少、绿色有多少、蓝色有多少，每个数都在 0 到 255 之间。纯红是 (255, 0, 0)；白色是 (255, 255, 255)；黑色是 (0, 0, 0)。

一张宽 4000、高 3000 个小方块的照片有 1200 万个像素。每个像素三个数，加起来就是排成一行的 3600 万个数。

排成一行的一列数叫**向量**，一共有多少个数，就是它的**维数**。在电脑眼里，你的照片就是一个 3600 万维的向量。它对照片做的一切，都是拿这列数做算术。



## 点积：乘一乘，再加起来

|                                                                                     |                 |                    |
|-------------------------------------------------------------------------------------|-----------------|--------------------|
|    | 30 个苹果 × 0.50 元 | = 15               |
|    | 12 根香蕉 × 0.25 元 | = 3                |
|  | 18 个橙子 × 1 元    | = 18               |
| <hr/>                                                                               |                 |                    |
|                                                                                     |                 | 15 + 3 + 18 = 36 元 |

**逐个相乘再相加：这就是点积。**

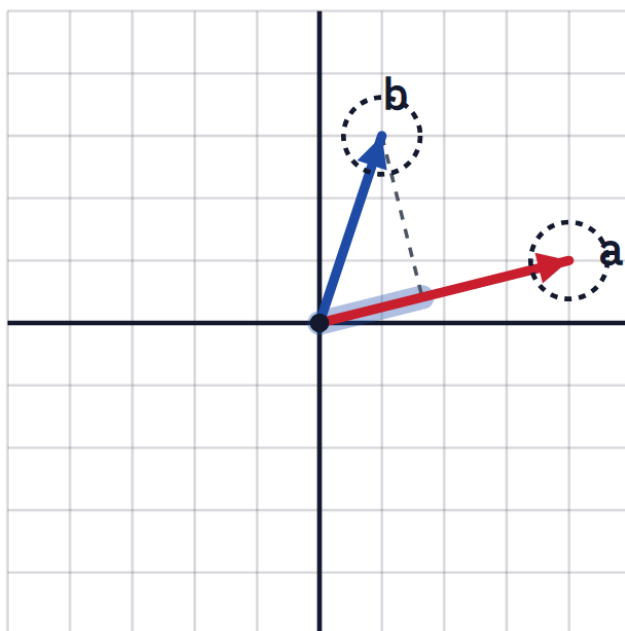
一个水果摊一天卖了 30 个苹果、12 根香蕉和 18 个橙子，单价分别是 0.50 元、0.25 元和 1 元。一共卖了多少钱？把每样的数量乘以单价，再加起来：15 + 3 + 18 = 36 元。两列数这样“逐个相乘再相加”，就是**点积**，也是电脑处理照片时做得最多的一道算术。这个例子来自一本数学教材，OpenStax。

用箭头来看，点积说的是另一件事：两支箭头有多指向同一侧。斜着推一辆购物车，只有朝车子前进方向的那部分力才算数。两支箭头指向同一侧，点积为正；互成直角，为零；方向相反，为负。

**笨蛋之王说：**  
乘一乘，加一加，就这么多。靠这个我能比价格、比箭头、比照片，还剩时间睡个午觉。

## 动手试试：两支箭头

拖动箭头的尖端，或者用滑块、键盘方向键来移动。找一个让点积等于 0 的位置：这时两支箭头互成直角。蓝色宽带是  $b$  在  $a$  上的影子：两支箭头越同向，影子越长。



a, 横向

4 a, 纵向  1

b, 横向

1 b, 纵向  3

点积  $a \cdot b$   $4 \cdot 1 + 1 \cdot 3 = 7$

长度  $\|a\| = 4.12 \cdot \|b\| = 3.16$

夹角  $57.5^\circ$

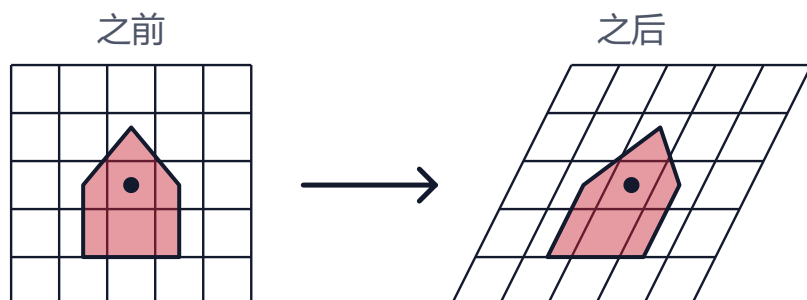
余弦（方向有多像）0.54

$a = (4, 1)$ ,  $b = (1, 3)$ ,  $a \cdot b = 7$ 。指向同一侧：点积为正。



互动小工具。在网站上试一试：<https://kingofdumbs.com/zh/lessons/vectors-and-matrices>

## 矩阵会挪动平面



**矩阵决定每个点去哪里。**  
直线还是直线，中心不动。

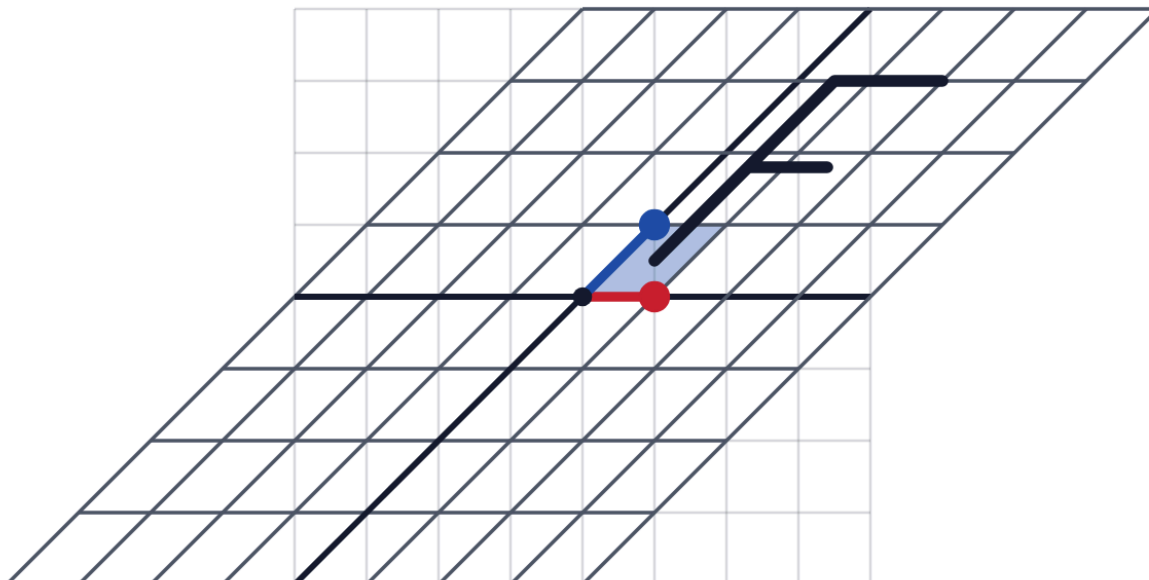
现在水果摊老板要比较两家店，每家价格不同。他把价格写成一张表：每家店一行，每种水果一列。这张表就是**矩阵**。想知道在每家店要花多少钱，就把购物清单和每一行逐个相乘再相加：每家店一个点积，一次全算出来。

矩阵还能挪动图画。想象一张画在橡皮膜上的网格：矩阵决定每个点去哪里，就像把橡皮膜拉长或推斜，但有两条规矩：直线还是直线，中心不动。

旋转或拉伸照片的程序，就是做这类计算来决定每个像素该去哪里。有些矩阵会把整幅图压成一条线：压扁以后，就再也还原不回来了。

## 动手试试：让网格变形

调一调矩阵的四个数，或者试试下面的按钮。红色箭头是第一列，即  $(1, 0)$  去的地方；蓝色箭头是第二列。找一个能把网格压成一条直线的矩阵：面积会变成 0。再看看镜像时字母 F 会怎样。



试一个

什么也不做

拉伸

转 90°

推斜

镜像

压扁

a (左上)



1

b (右上)



1

c (左下)



0

d (右下)



1

(1, 0) 变成 (1, 0)

(0, 1) 变成 (1, 1)

面积乘以 (ad - bc) **1**

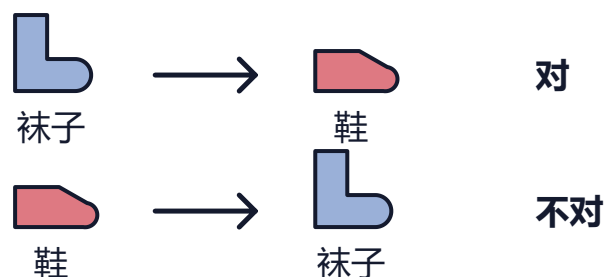
剩下的维数 2: 平面还是平面

矩阵  $\begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$ : 面积乘以 1, 剩下 2 维。



互动小工具。在网站上试一试: <https://kingofdumbs.com/zh/lessons/vectors-and-matrices>

## 矩阵乘法：顺序很重要



接连做两个矩阵就是把它们相乘，  
顺序很重要。

穿衣服要先穿袜子再穿鞋，反过来就不是一回事了。矩阵也一样：先做一个变换再做另一个，相当于一个矩阵，这个矩阵是把两个矩阵**相乘**得到的，而且顺序很重要。先旋转照片再横向拉伸，和先拉伸再旋转，结果不一样。

开头那个问题的答案就在这里。一个认猫的神经网络，把照片当成一个巨大的向量接收进来，再一个接一个地乘上矩阵，中间还夹着几个别的步骤。这些矩阵的每一行都像一张价目表：它和照片的点积得出一个数。每张照片都要做上百万次这样的乘加运算。

所以显卡特别适合人工智能：它就是为了同时做大量乘法而设计的。

### 笨蛋之王说：

我看不见猫，只看见 3600 万个数。难的是知道该拿哪些矩阵去乘它们。

## 最后挑战

1. 一张  $100 \times 100$  像素的彩色照片。电脑打开它时要存多少个数？

- a. 100
- b. 10000，每个像素一个
- c. 30000：每个像素三个，红、绿、蓝

## 2. 两支箭头互成直角。它们的点积是多少？

- a. 1
- b. 0
- c. 要看它们有多长

## 3. 把照片转 $90^\circ$ ，再横向拉伸。这和先拉伸、再旋转一样吗？

- a. 一样：乘法怎么换顺序都一样
- b. 不一样：矩阵相乘，顺序很重要
- c. 只有照片是正方形时才一样



## 三句话总结

1. 在电脑眼里，照片是一列数（每个像素三个，取值 0 到 255）：一个向量，它的维数就是数的个数。
2. 点积把两列数逐项相乘再相加：既能算出总价，就像水果摊的账；也能说明两支箭头有多同向（垂直时为 0）。
3. 矩阵是一张能挪动平面上每个点的数表，矩阵乘法把这些变换串起来，而且顺序要紧：这正是神经网络对每张照片做上百万次的运算。



## 参考资料

- 向量、长度和点积：OpenStax, *Calculus Volume 3*, [2.1](#) 和 [2.3](#)（水果摊的例子，定理 2.4 和 2.5）。矩阵及其乘法：OpenStax, *College Algebra 2e*, [7.5](#)。
- 变换、行列式和维数：D. Austin, *Understanding Linear Algebra*, [2.2](#)、[2.5](#)、[2.6](#)、[3.4](#) 和 [3.5](#)。
- 像素和图像变换：Pillow 13.0, [Concepts](#) 和 [ImageTransform](#)。从照片到张量：torchvision 0.29, [ToTensor](#)。
- 神经网络：PyTorch 2.14, [nn.Linear](#) 和 [cosine\\_similarity](#)。NVIDIA, [Matrix Multiplication Background](#) 和 [Linear/Fully-Connected Layers](#)。

**\*\*这一课是怎么做的：\*\***在 Claude 的帮助下，根据上面的资料调研和撰写，资料查阅于 2026 年 10 月 17 日。没有单独链接的数字，是根据定义算出来的。例子（4000 × 3000 的照片、两家店、有 1000 个输出的层）都是编的。插图都是原创的。

## 答案

1. **c)** 30000: 每个像素三个, 红、绿、蓝. 对。  $100 \times 100 \times 3$ : 一个 30000 维的向量。
2. **b)** 0. 答对了。互相垂直, 点积就是 0, 不管多长。
3. **b)** 不一样: 矩阵相乘, 顺序很重要. 没错。就像袜子和鞋。

---

King of Dumbs 的课程, kingofdumbs.com。这是离线版; 可以动手玩的原版在:

<https://kingofdumbs.com/zh/lessons/vectors-and-matrices>

更新于 2026年10月17日.