

# Data Mining HW1 Report

r08725021 王鼎元 r08725023 何昱辰 r08725025 陳禹媛 r08725026 林楷翊

## 研究動機與目的

在機器學習的領域中，降維處理對網路大小有著不可忽視的影響。我們找到了一個有關狗的圖片資料集，於是決定運用上課所教的降維方式，看看是否能對圖片分類的問題有所幫助。

## 資料描述

這個資料集來自 Stanford 大學，其中總共有 20580 張有標記、大小不一的彩色圖片，並依狗的種類分為 120 個種。每張圖片都對應著一個 annotation 檔案，裡面詳細記錄著這張圖片的大小、狗的品種、狗在圖片中出現的範圍 (bounding box)。

- 圖片



■ annotation

```
<annotation>
  <folder>02085620</folder>
  <filename>n02085620_10074</filename>
  <source>
    <database>ImageNet database</database>
  </source>
  <size> <!-- Image size. -->
    <width>333</width>
    <height>500</height>
    <depth>3</depth>
  </size>
  <segment>0</segment>
  <object>
    <name>Chihuahua</name> <!-- Breed. -->
    <pose>Unspecified</pose>
    <truncated>0</truncated>
    <difficult>0</difficult>
    <bndbox> <!-- Bounding box. -->
      <xmin>25</xmin>
      <ymin>10</ymin>
      <xmax>276</xmax>
      <ymax>498</ymax>
    </bndbox>
  </object>
</annotation>
```

# 實驗方法

## 資料前處理

本資料集最初的結構是先分成 annotations、images 兩大資料夾，下面再將每個分類獨立成各個資料夾，如下：

```
stanford-dogs-dataset/  
├── annotations  
│   └── Annotation  
│       ├── n02085620-Chihuahua  
│       │   ├── n02085620_10074  
│       │   ├── n02085620_10131  
│       │   └── ...  
│       ├── n02085782-Japanese_spaniel  
│       │   ├── n02085782_1039  
│       │   └── n02085782_1058  
│       └── ...  
└── images  
    └── Images  
        ├── n02085620-Chihuahua  
        │   ├── n02085620_10074.jpg  
        │   ├── n02085620_10131.jpg  
        │   └── ...  
        ├── n02085782-Japanese_spaniel  
        │   ├── n02085782_1039.jpg  
        │   └── n02085782_1058.jpg  
        └── ...
```

為方便資料讀取，我們先將資料依 0.7、0.15、0.15 的比例分成訓練集、驗證集與測試集，再整理成只有 train、valid、test 三個資料夾，且將每張圖及其對應標記整合成 .h5 檔，如下：

```
dogdata/  
├── test  
│   ├── n02085620_11948.h5  
│   ├── n02085620_1235.h5  
│   └── ...  
├── train  
│   ├── n02085620_10074.h5  
│   ├── n02085620_10131.h5  
│   └── ...  
└── valid  
    ├── n02085620_12718.h5  
    ├── n02085620_1455.h5  
    └── ...
```

## 資料降維

我們主要採用三種降維的方式，分別是 DWT (discrete wavelet transform)、SVD (singular value decomposition)、NMF (non-negative matrix factorization)。

### 1. Discrete wavelet transform

我們對原本的圖片做了 2 維的 Haar wavelet transform，並得到 sum-sum 的 approximation 做為我們的降維結果。

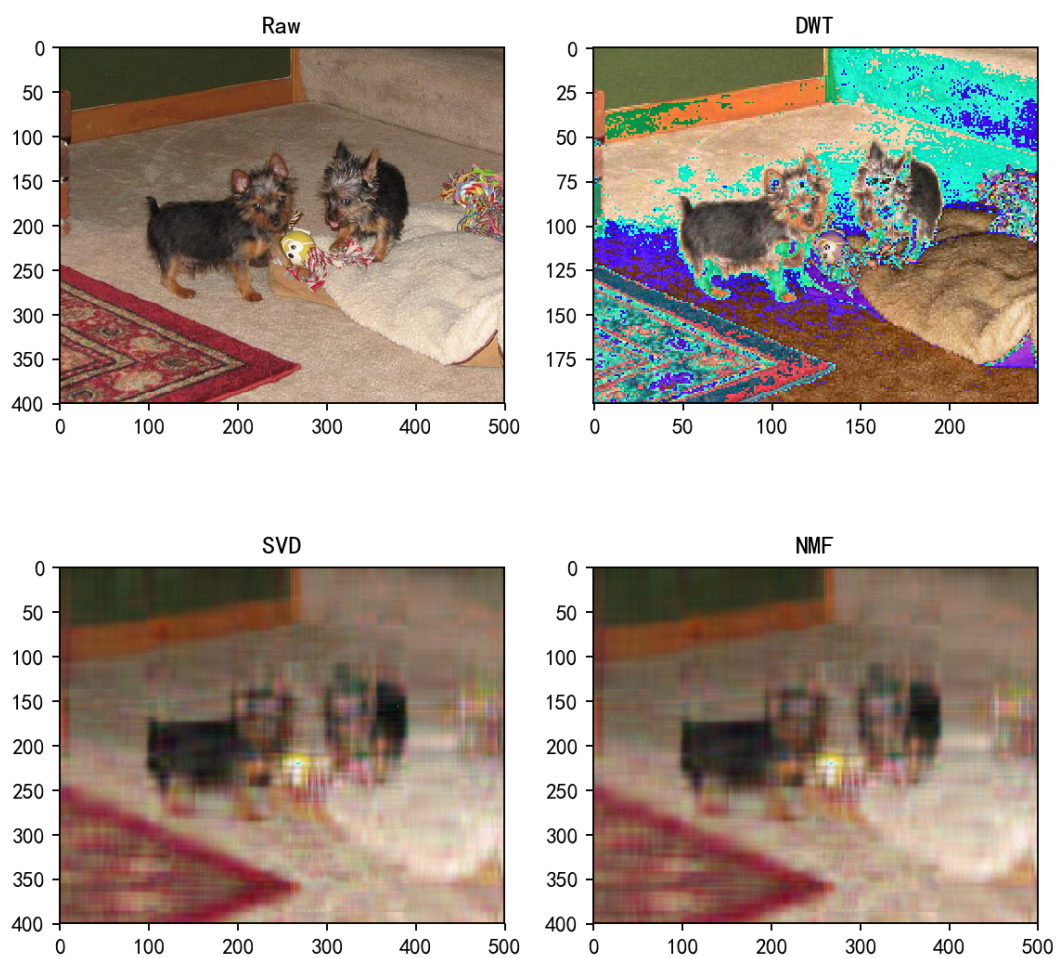
### 2. Singular value decomposition

利用 SVD 分解出  $U$ 、 $\Lambda$ 、 $V^T$ ，並取出前 10 個 components 來重建圖片做為我們的降維結果。

### 3. Non-negative matrix factorization

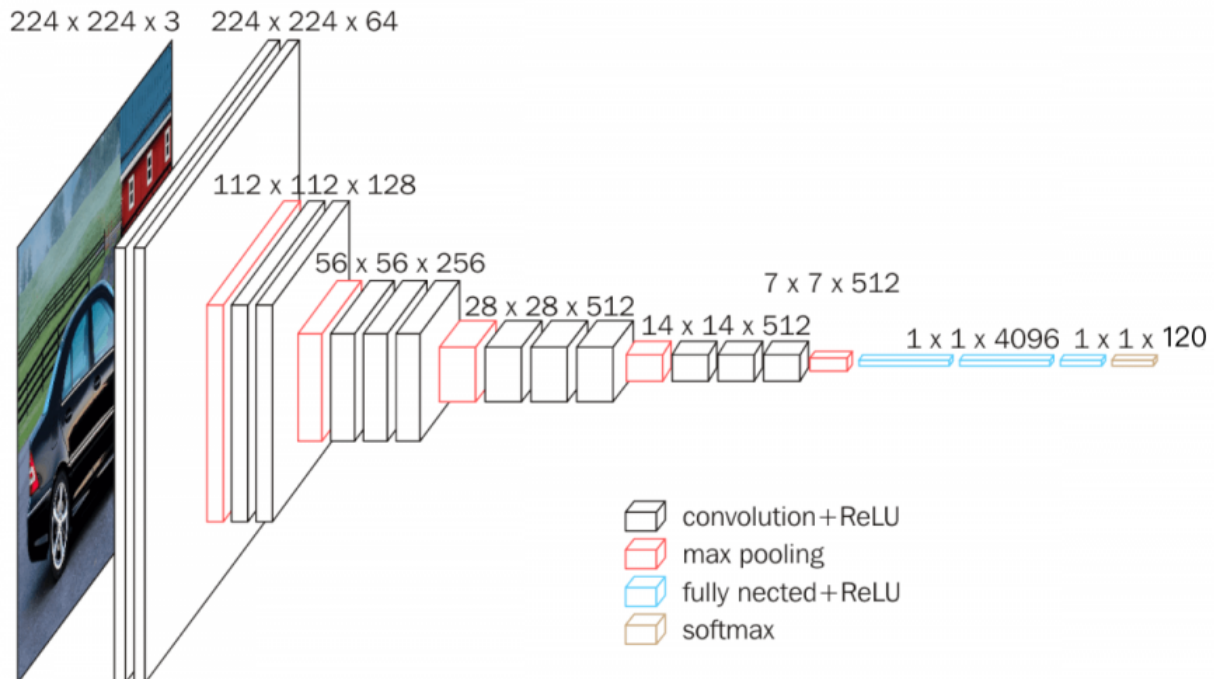
利用 NMF 分解出  $W$ 、 $H$ ，其中  $W$  的維度為  $(N \times 10)$ ， $V$  的維度為  $(10 \times M)$ ，並用  $W \times H$  來重建圖片做為我們的降維結果。

下圖為降維前與各個降維方法的結果：



## 機器學習分類

在前處理與降維後，我們將資料餵進一個卷積神經網路 (CNN) 中並求得分類結果。我們使用預訓練模型 VGG-16 作為基本架構，並利用目標資料來 fine-tune。



上圖為 VGG-16 的網路架構。圖片來源：<https://neurohive.io/en/popular-networks/vgg16/>

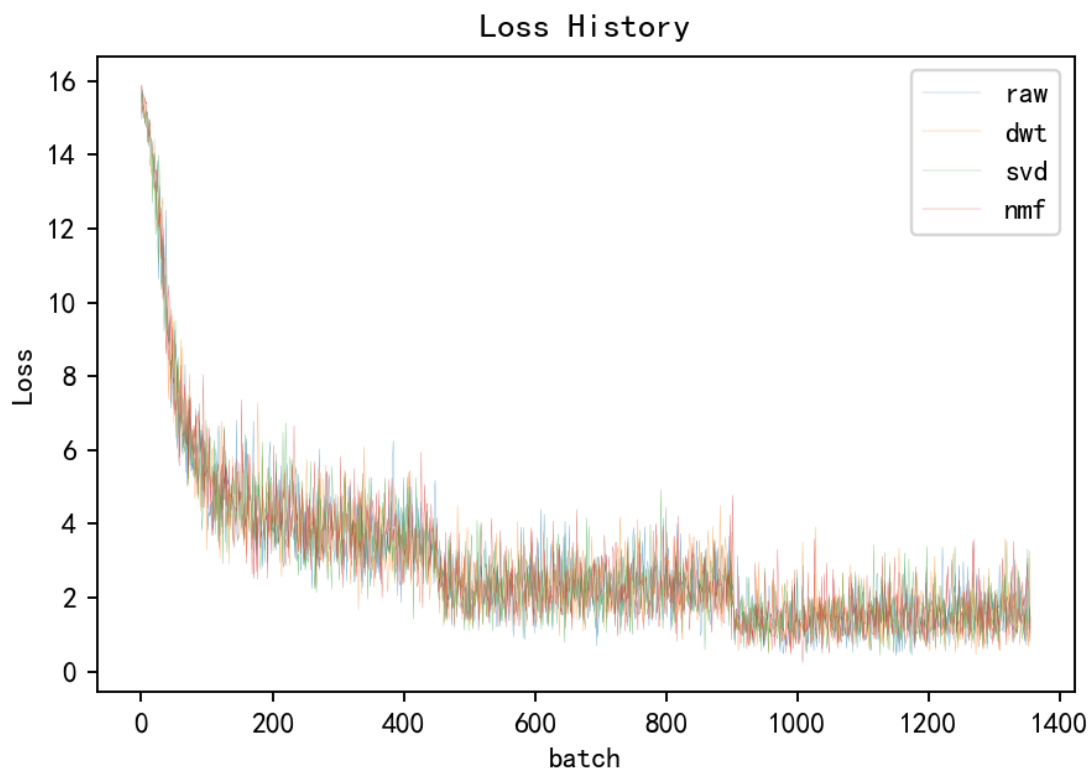
## 實驗結果與分析

實驗結果如下表所示：

Dimension reduction method	Accuracy
Raw (沒有經過降維)	0.7557024793388429
DWT	0.7464462809917355
SVD	0.7375206611570247
NMF	0.7289256198347107

由上表可知，經過降維後分類的準確度都下降了，我們推論因為降維會使資料的特徵減少，導致 CNN 少抓取了一些可能有用的資訊。而相較於 SVD、NMF，DWT 的降維方法更能夠保留訊息。

下圖為每個降維方法的訓練曲線：



由上圖可知，雖然資料降維會使一些有用的資訊消失，但整體的訓練過程仍然相當快收斂，使用原圖跟降維圖片的差異在訓練時並不大，故可推論降維後的圖片仍能保有一定程度的資訊量，故在較大的模型訓練時應該能有效的降低訓練參數量。