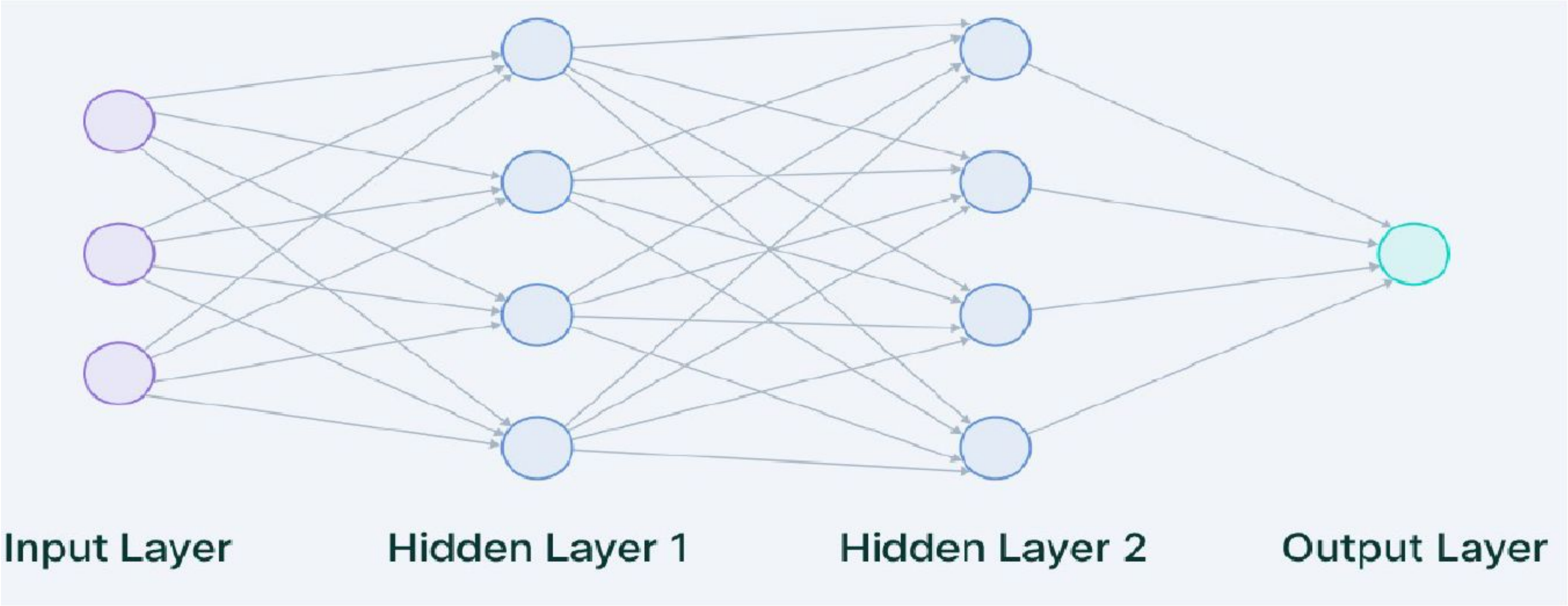
Introduction to Deep Learning

- Machine learning requires data preprocessing, which involves human intervention.
- The neural networks in deep learning are capable of extracting features; hence no human intervention is required.
- Deep Learning can process unstructured data.
- Deep Learning is usually based on representative learning i.e., finding and extracting vital information or patterns that represent the entire dataset.
- Deep learning is computationally expensive and time-consuming.

How does Deep Learning work?

- Deep Neural Networks have multiple layers of interconnected artificial neurons or nodes that are stacked together.
- Each of these nodes has a simple mathematical function - usually a linear function that performs extraction and mapping of information.
- There are three layers to a deep neural network: the input layer, hidden layers, and the output layer.

How does Deep Learning work?



Types of Neural Network

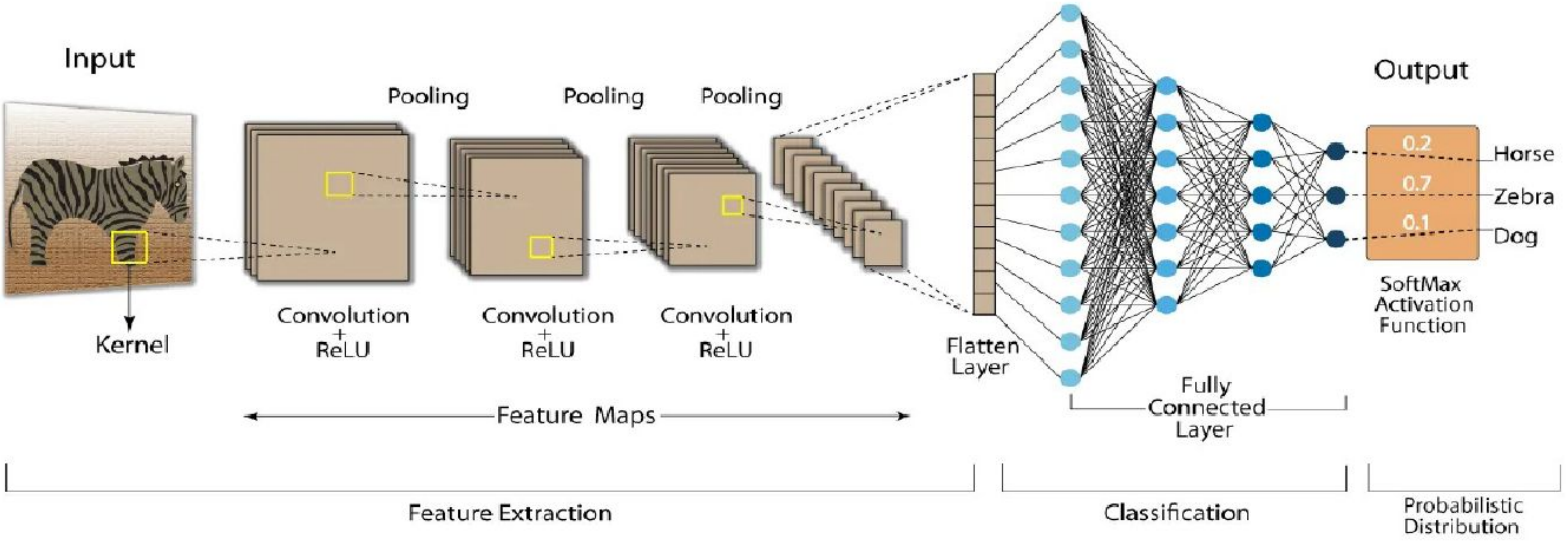
- Artificial Neural Network
- Convolutional Neural Network
- Recurrent Neural Network
- Generative Adversarial Network

Deep learning features with CNN

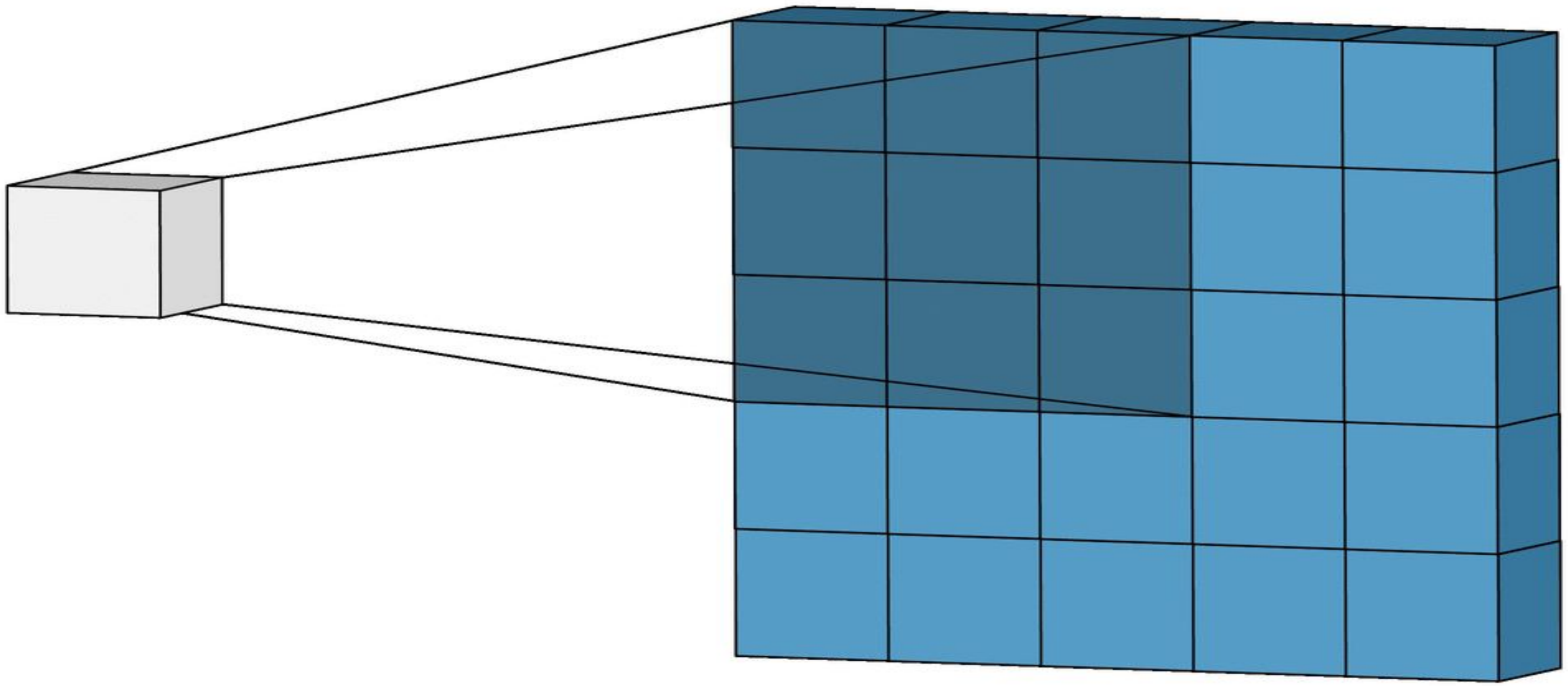
CNN

- The Convolutional Neural Networks or CNNs are primarily used for tasks related to computer vision or image processing.
- CNNs are extremely good in modeling spatial data such as 2D or 3D images and videos.
- They can extract features and patterns within an image, enabling tasks such as image classification or object detection.

CNN


Input
Pooling
Pooling
Pooling
Output
Kernel
Convolution + ReLU
Convolution + ReLU
Convolution + ReLU
Flatten Layer
Fully Connected Layer
0.2
0.7
0.1
Horse
Zebra
Dog
SoftMax Activation Function
Feature Maps
Feature Extraction
Classification
Probabilistic Distribution

Convolution & Pooling Layers



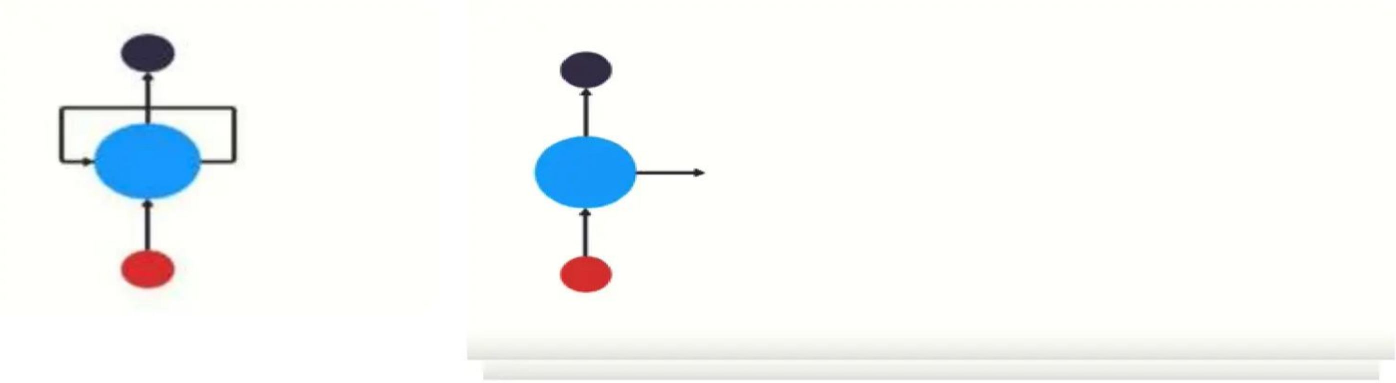
Overall Process:

1. Input image passes through the convolution layer, applying filters to extract features.
2. The resulting feature map undergoes an activation function (e.g., ReLU) to introduce non-linearity.
3. Max-pooling layers down sample the feature map by selecting maximum values in local regions.
4. This process of convolution and pooling is typically repeated in a stack of layers to create a deep CNN architecture.

Convolution layers with filters extract various features, such as edges, textures, and patterns. Max-pooling reduces spatial dimensions and retains essential information. Together, these layers enable CNNs to learn hierarchical representations of data and are critical for their success in computer vision tasks.

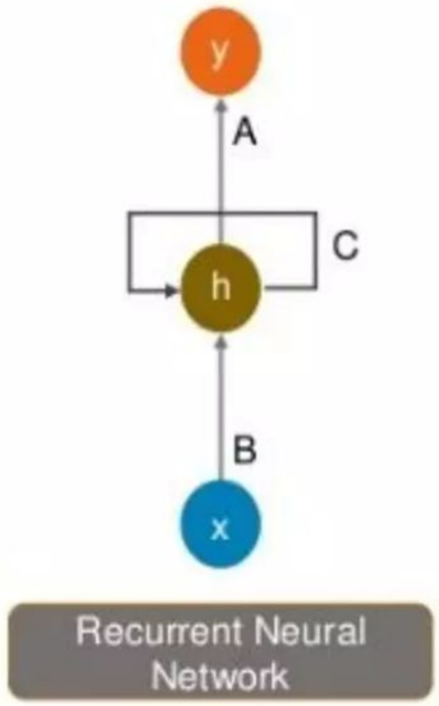
Deep learning features with LSTM

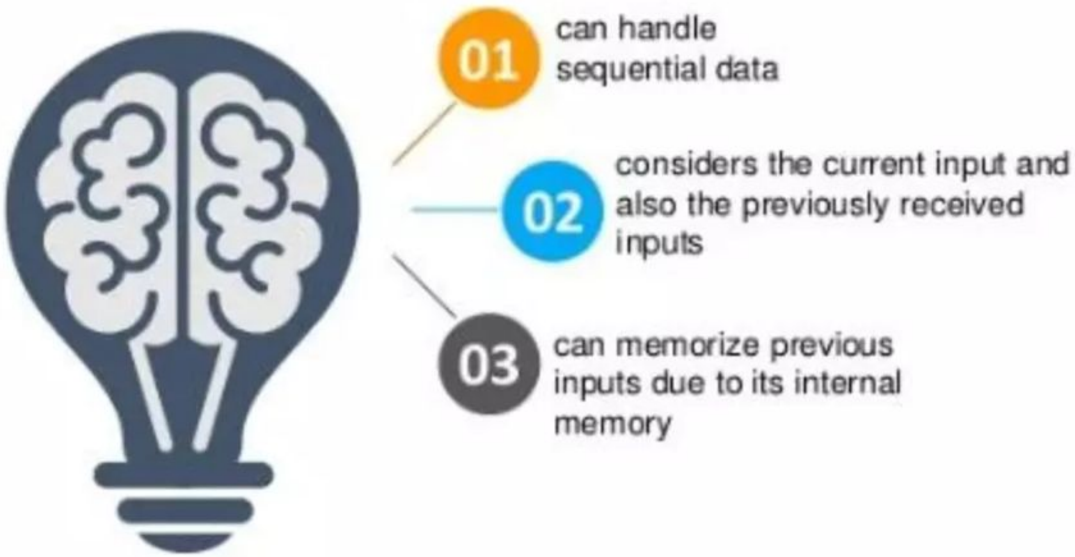
Recurrent Neural Networks



Recurrent Neural Network

- Recurrent Neural Network works on the principle of saving the output layer and feeding this back to the input in order to predict the output of the layer.

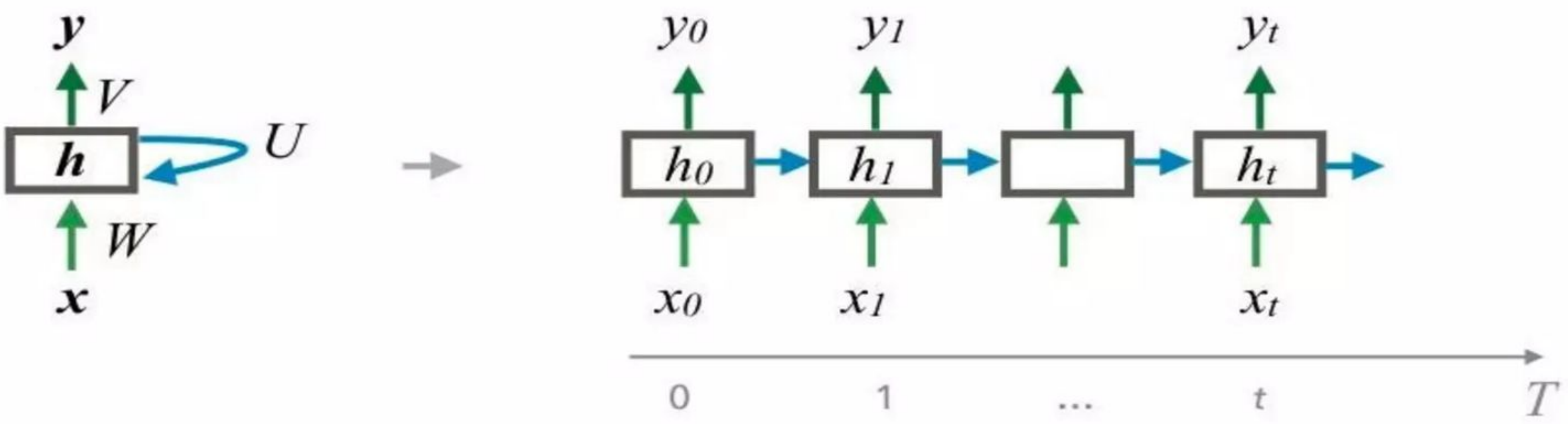




- Recurrent Neural Network is basically a generalization of feed-forward neural network that has an internal memory. RNNs are a special kind of neural networks that are designed to effectively deal with sequential data. This kind of data includes time series (a list of values of some parameters over a certain period of time) text documents, which can be seen as a sequence of words, or audio, which can be seen as a sequence of sound frequencies over time.
- RNN is recurrent in nature as it performs the same function for every input of data while the output of the current input depends on the past one computation. For making a decision, it considers the current input and the output that it has learned from the previous input.
- Cells that are a function of inputs from previous time steps are also known as memory cells.
- Unlike feed-forward neural networks, RNNs can use their internal state (memory) to process sequences of inputs. In other neural networks, all the inputs are independent of each other. But in RNN, all the inputs are related to each other.

Recurrent Neural Network : Intuition

- Recurrent Neural Network (RNN) is a neural network model proposed in the 80's for modelling time series.
- The structure of the network is similar to feedforward neural network, with the distinction that it allows a recurrent hidden state whose activation at each time is dependent on that of the previous time (cycle).

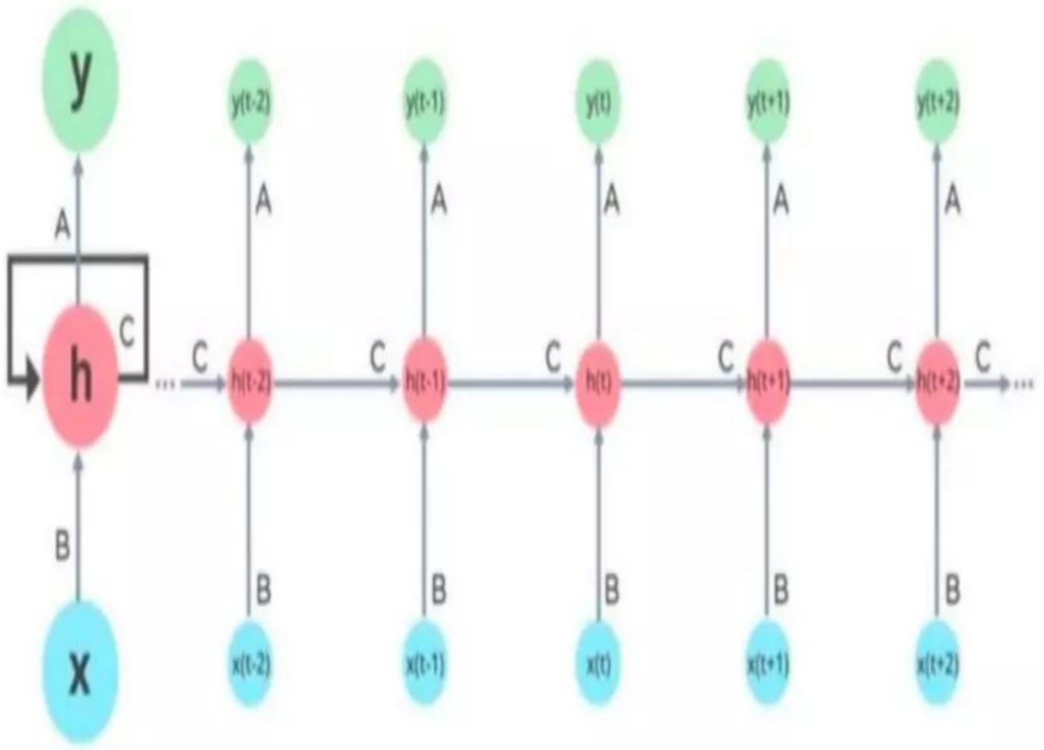


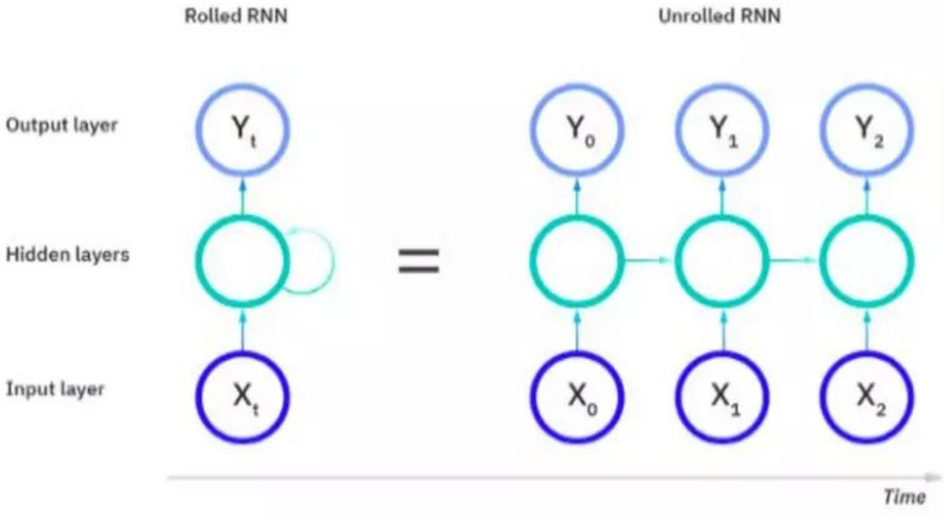
$$\mathbf{y}_{(t)} = \phi(\mathbf{x}_{(t)}{}^T \cdot \mathbf{w}_x + \mathbf{y}_{(t-1)}{}^T \cdot \mathbf{w}_y + b)$$

ϕ() is the activation function like ReLU

bias

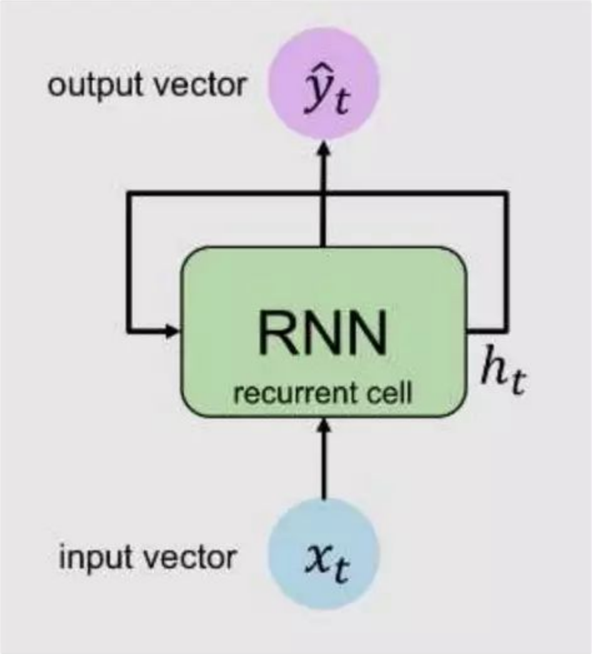
How does RNN look like?





Rolled RNN and Unrolled RNN (IBM)

Recurrent Neural Network

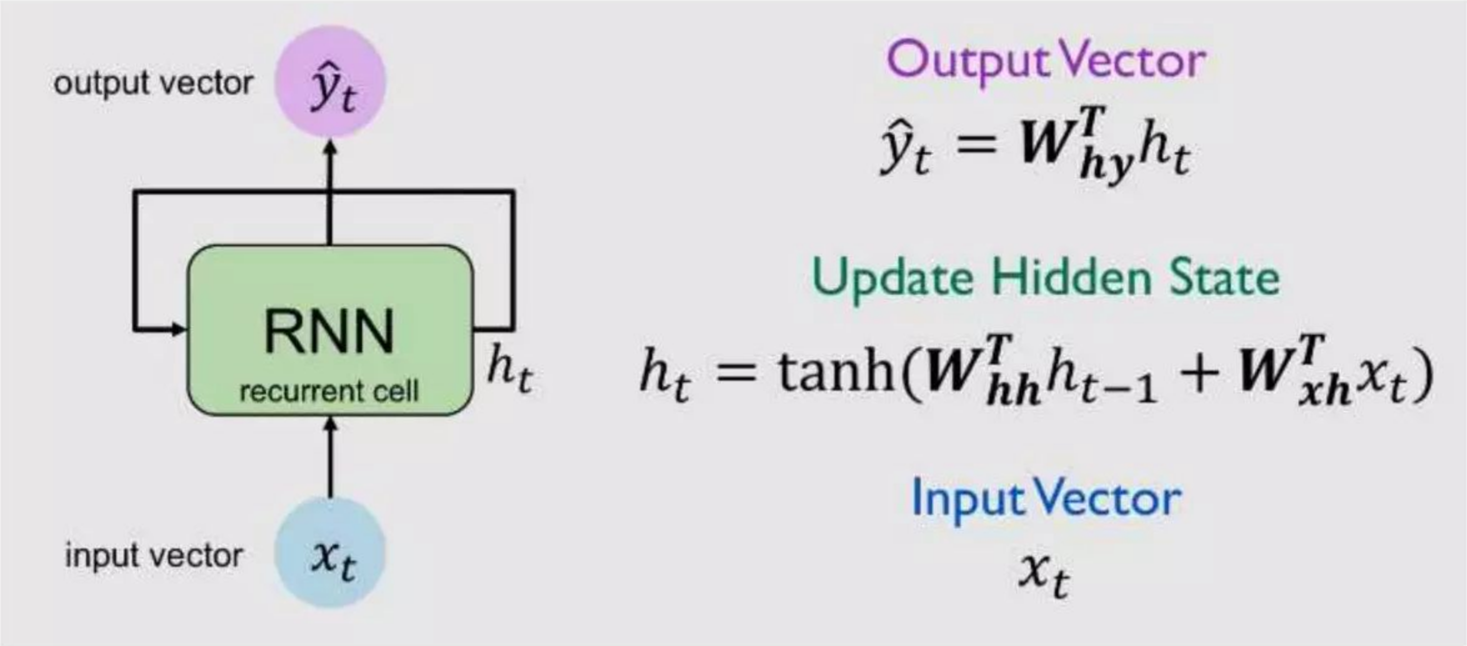


Apply a recurrence relation at every time step to process a sequence:

$$\underbrace{\boxed{h_t}}_{\text{cell state}} = \underbrace{\boxed{f_W}}_{\text{function parameterized by W}}(\underbrace{\boxed{h_{t-1}}}_{\text{old state}}, \underbrace{\boxed{x_t}}_{\text{input vector at time step } t})$$

Note: The same function and set of parameters are used at every time step

RNN State Update and Output



Deep learning features with LSTM

Sequence Data Modeling

Neural Network are being applied to the problems that involve sequential processing of data

Inferring and understanding genomics sequences

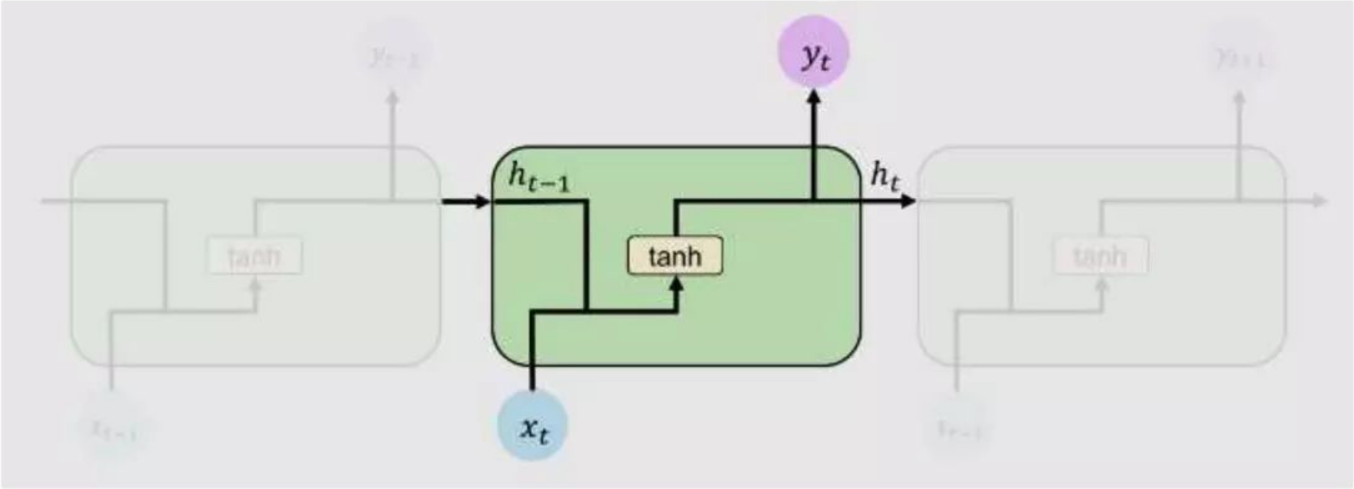
Sequence Modeling: Design Criteria

To model sequences, we need to:

1. Handle variable-length sequences
2. Track long-term dependencies
3. Maintain information about order
4. **Share parameters** across the sequence

Standard RNN

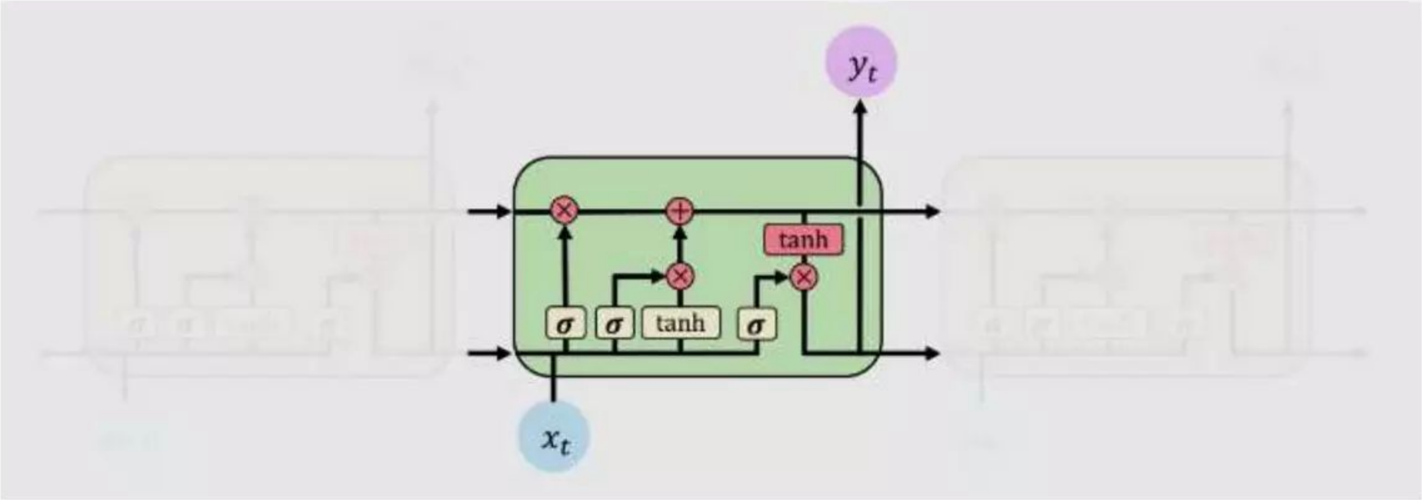
In a standard RNN, repeating modules contain a simple computation node



Long Short Term Memory (LSTMs)

LSTMs networks rely on gated cell to track information throughout many time steps.

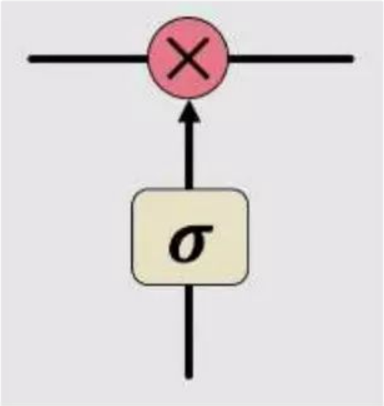
LSTM modules contain computational blocks that control information flow



LSTM cells are able to track information throughout many time steps

Long Short Term Memory (LSTMs)

Information is added or removed through structures called gates

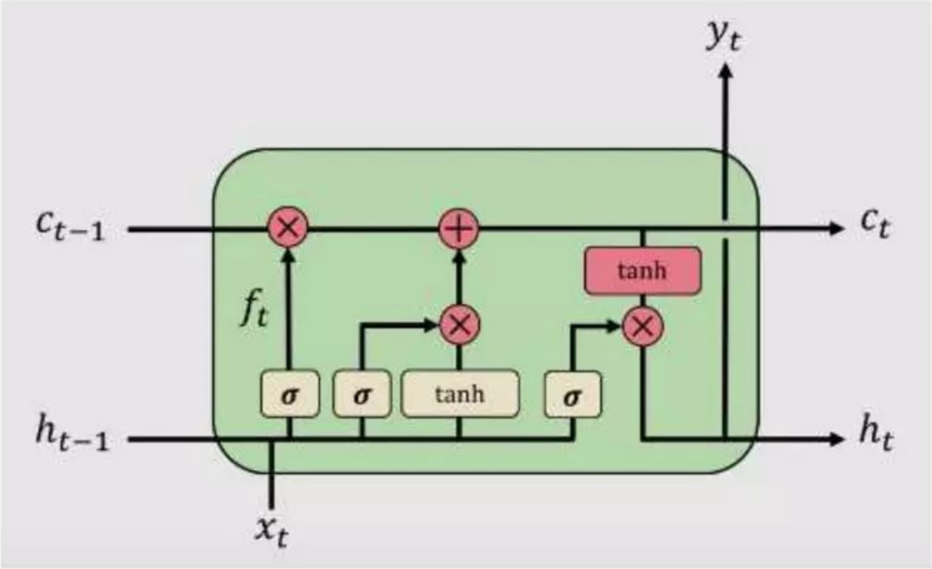


Gates optionally let information through, for example via a sigmoid neural net layer and point wise multiplication

Long Short Term Memory (LSTMs)

How does LSTMs work ?

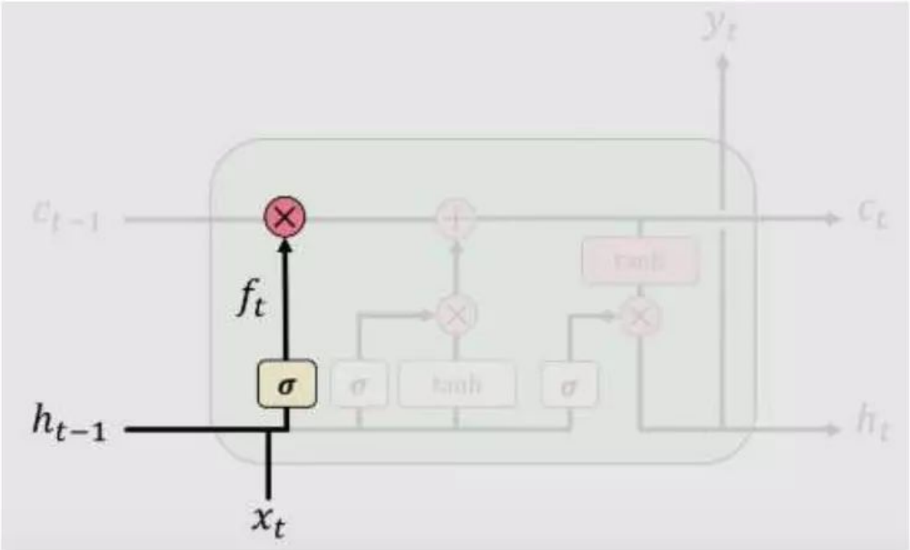
1) Forget 2) Store 3) Update 4) Output



Long Short Term Memory (LSTMs)

1) Forget 2) Store 3) Update 4) Output

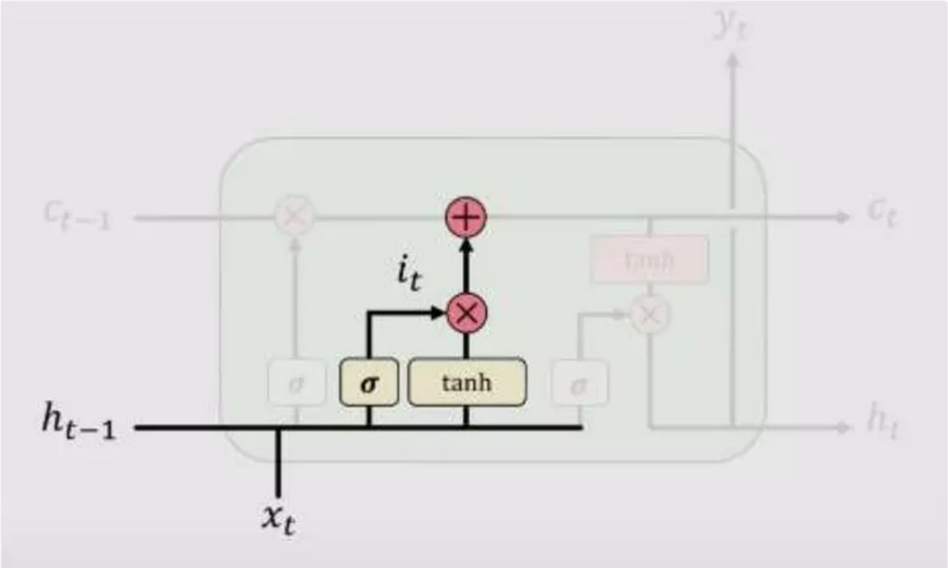
LSTMs forget irrelevant parts of the previous state



Long Short Term Memory (LSTMs)

1) Forget **2) Store** 3) Update 4) Output

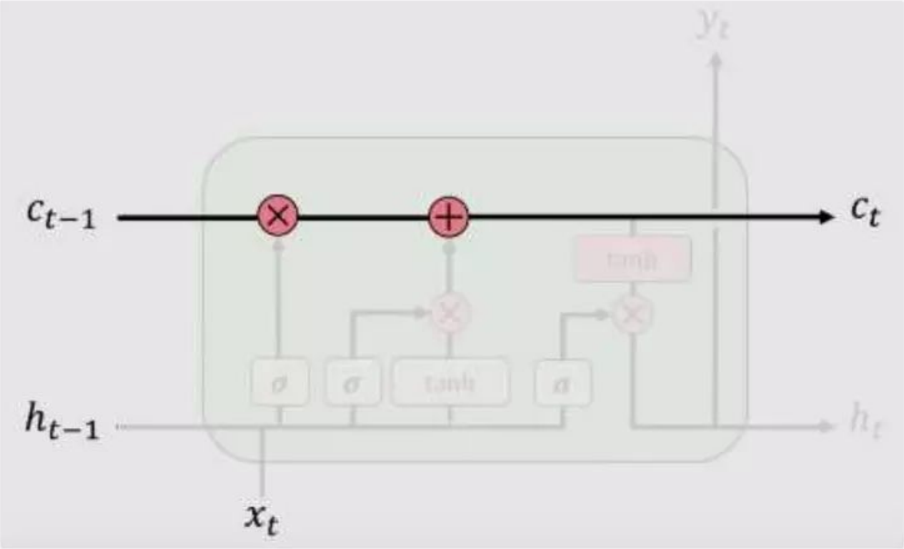
LSTMs store relevant new information into the cell state



Long Short Term Memory (LSTMs)

1) Forget 2) Store **3) Update** 4) Output

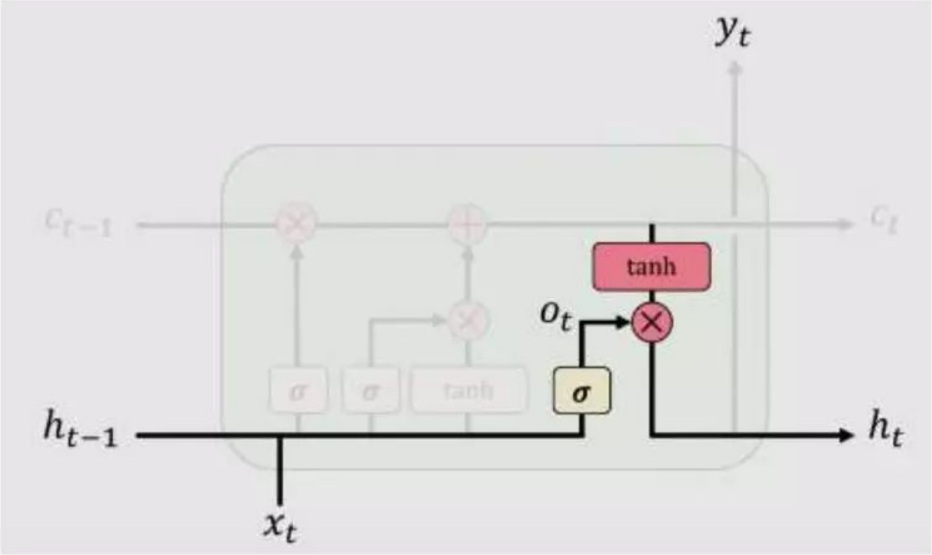
LSTMs selectively update cell state values



Long Short Term Memory (LSTMs)

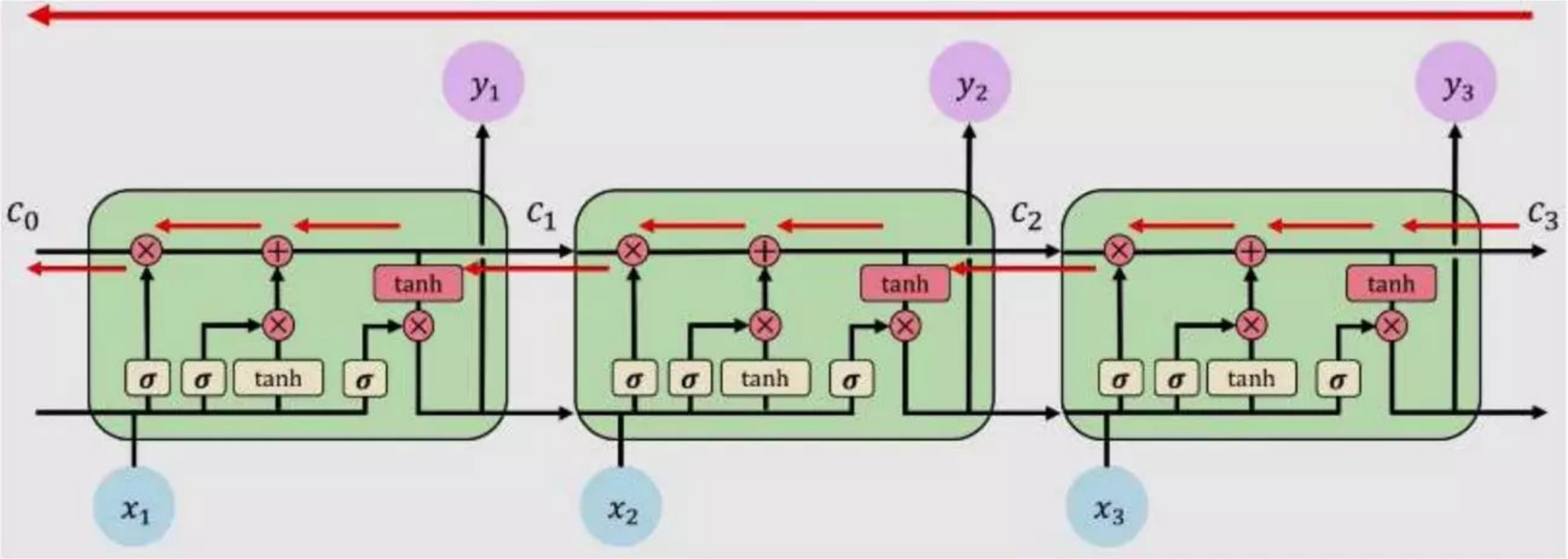
1) Forget 2) Store 3) Update **4) Output**

The output gate controls what information is sent to the next time step



LSTM Gradient Flow

Uninterrupted gradient flow!

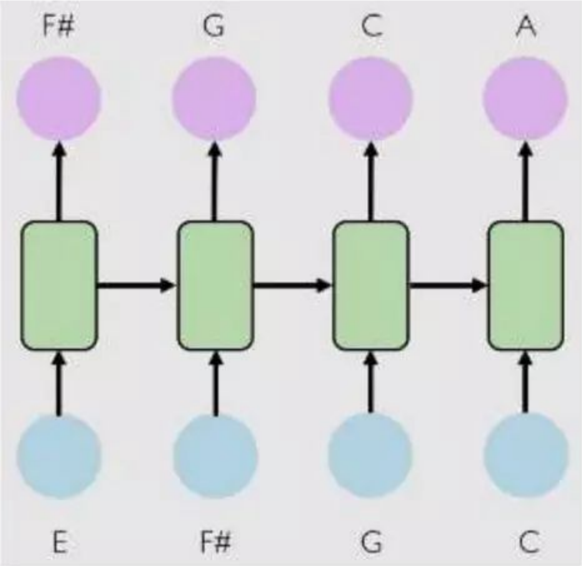


LSTMs: Key Concepts

1. Maintain a separate cell state from what is outputted
2. Use gates to control the flow of information
    - Forget gate gets rid of irrelevant information
    - Store relevant information from current input
    - Selectively update cell state
    - Output gate returns a filtered version of the cell state
3. Back propagation through time with uninterrupted gradient flow

Application of RNN

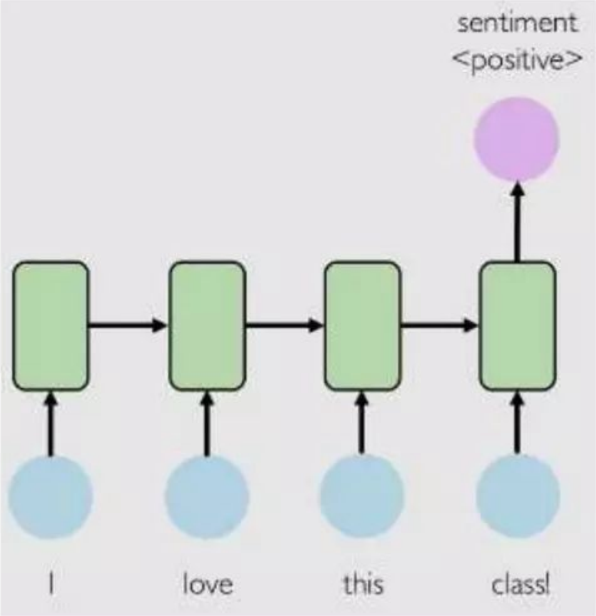
Music Generation



Sentiment Classification

Input: Sequence of words

Output: probability of having positive sentiment



Application of RNN

Trajectory Prediction: Self-Driving Cars

Environmental Modeling

Machine Translation

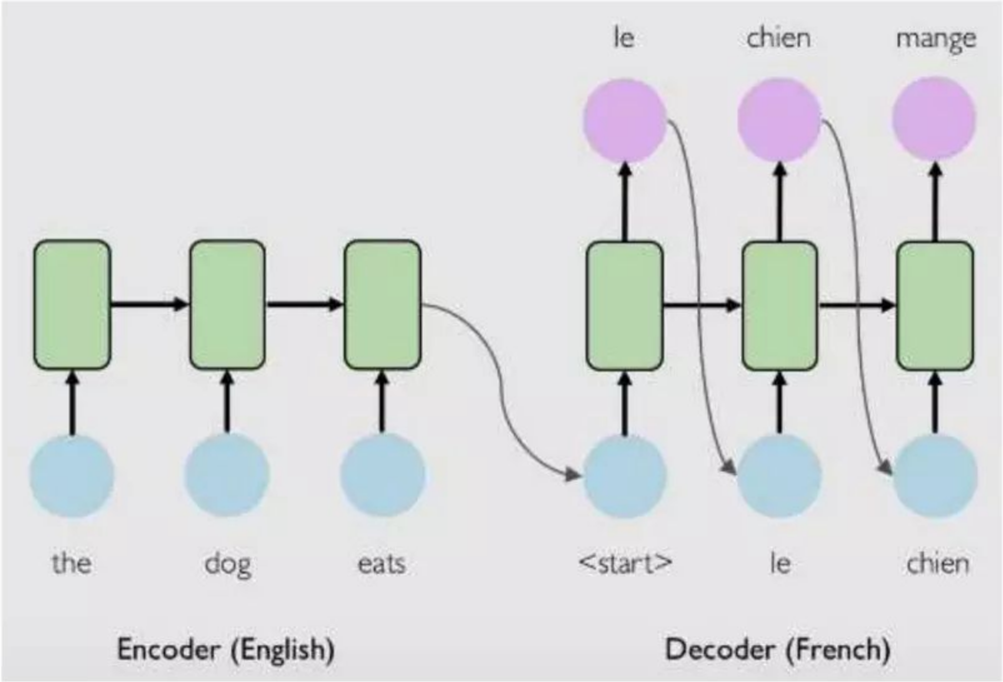
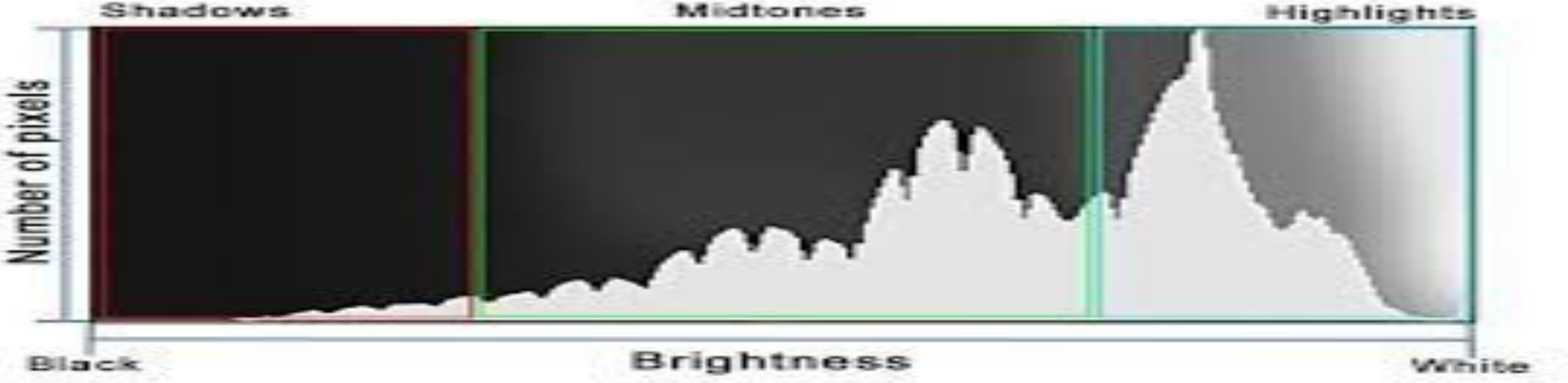


Image Histogram

Histogram

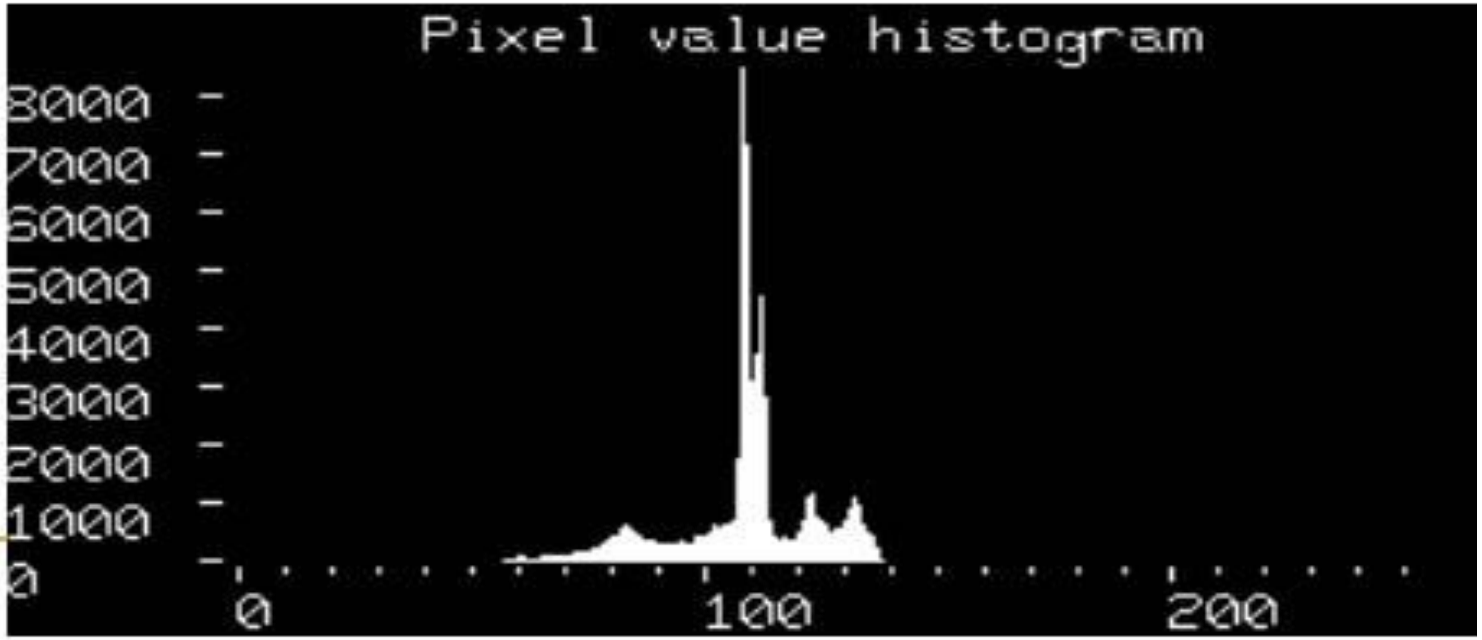
The (intensity or brightness) histogram shows how many times a particular grey level (intensity) appears in an image.

For example, 0 - black, 255 – white



Histogram (Cont)

An image has low contrast when the complete range of possible values is not used. Inspection of the histogram shows this lack of contrast.



Histogram of color images

RGB color can be converted to a gray scale value by

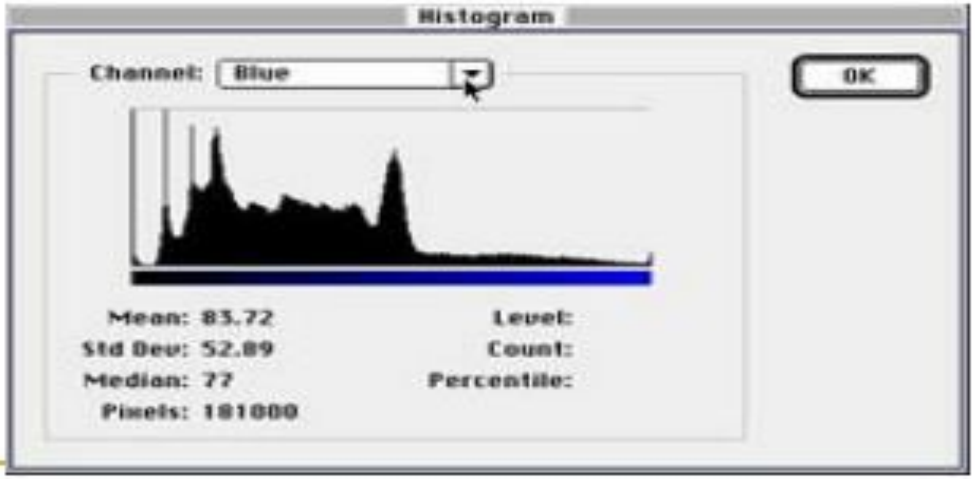
$$Y = 0.299R + 0.587G + 0.114B$$

Y: the grayscale component in the YIQ color space used in NTSC television.
The weights reflect the eye's brightness sensitivity to the color primaries.

Histogram of color images (Cont)

Histogram:

individual histograms of red, green and blue

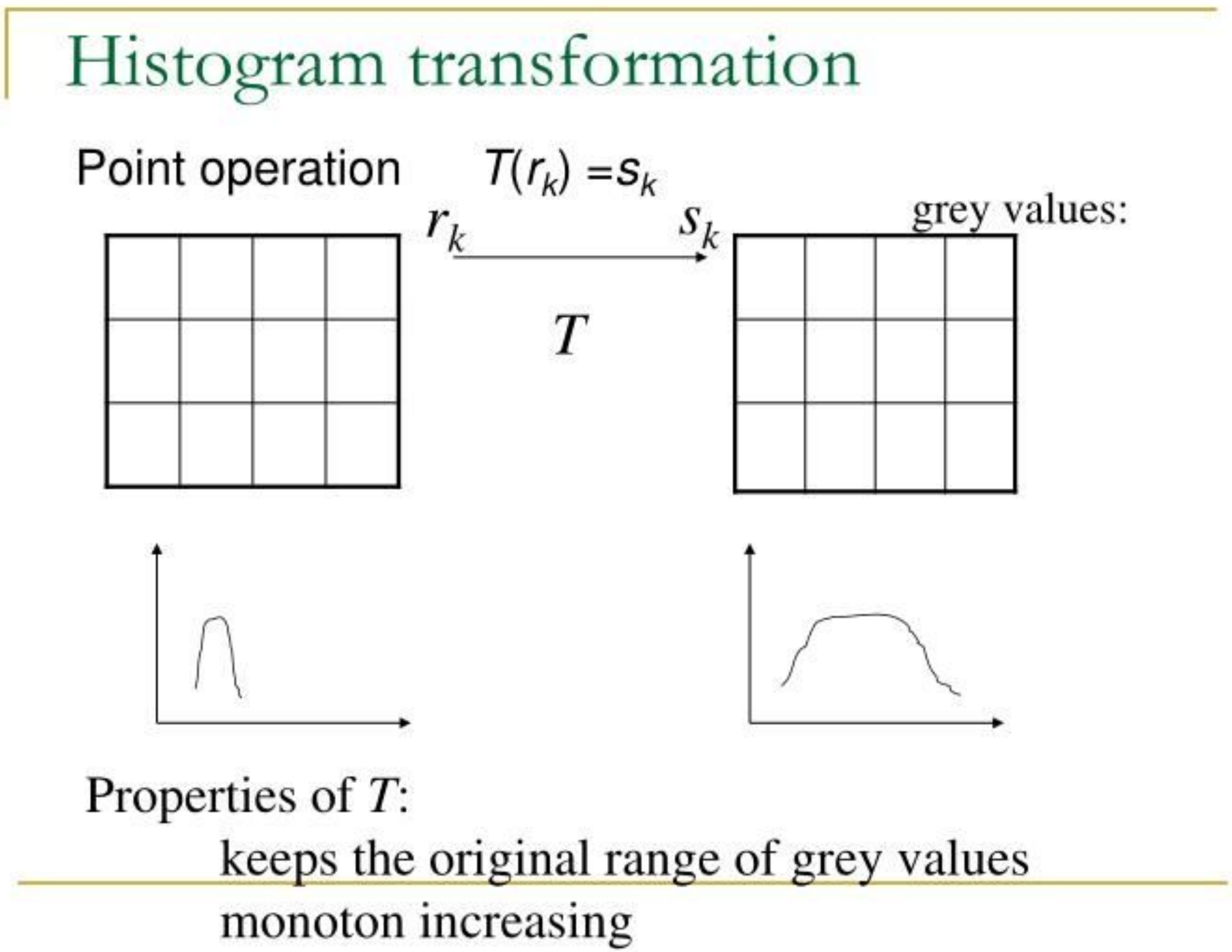


Blue

Histogram of color images (Cont)

or

a 3-D histogram can be produced, with the three axes representing the red, blue and green channels, and brightness at each point representing the pixel count

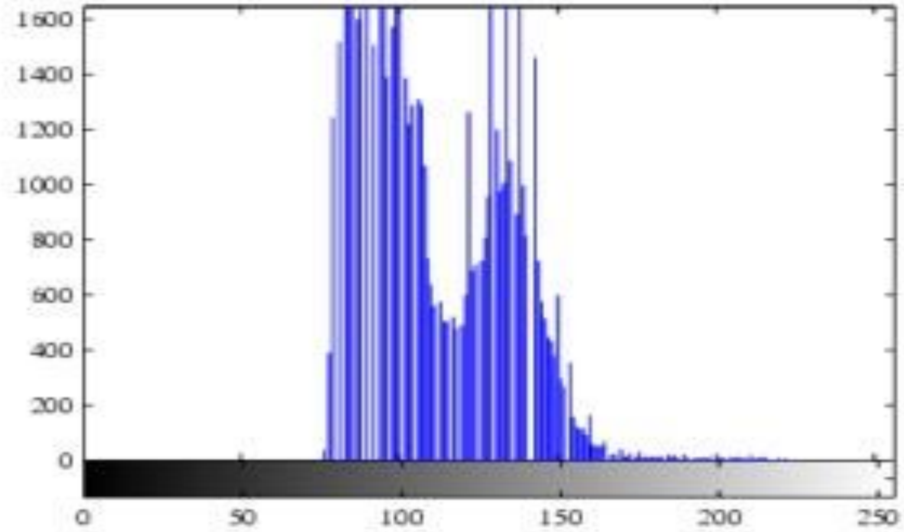


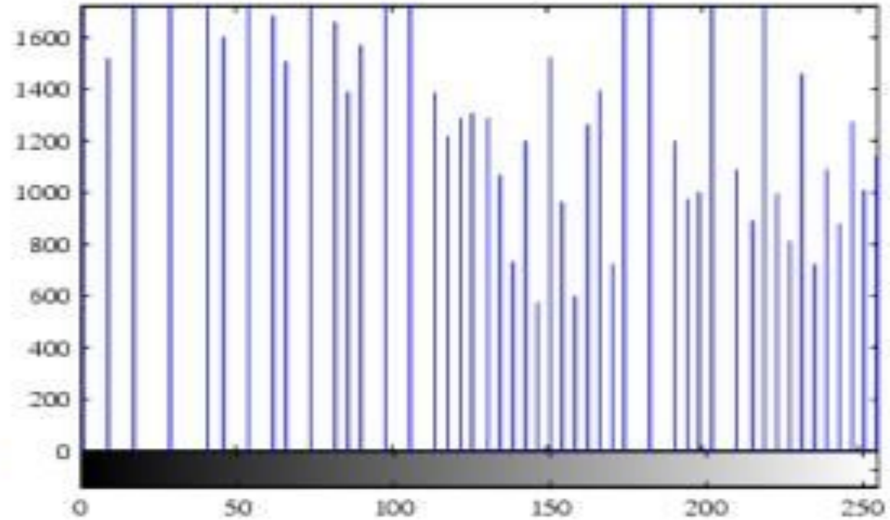
A histogram transformation function. is simply the cumulative probability distribution (i.e. cumulative histogram) of the original image. Thus, an image which is transformed using its cumulative histogram yields an output histogram which is flat! is the number of pixels at intensity level k or less.

Histogram Equalization (HE)

Transforms the intensity values so that the histogram of the output image approximately matches the flat (uniform) histogram

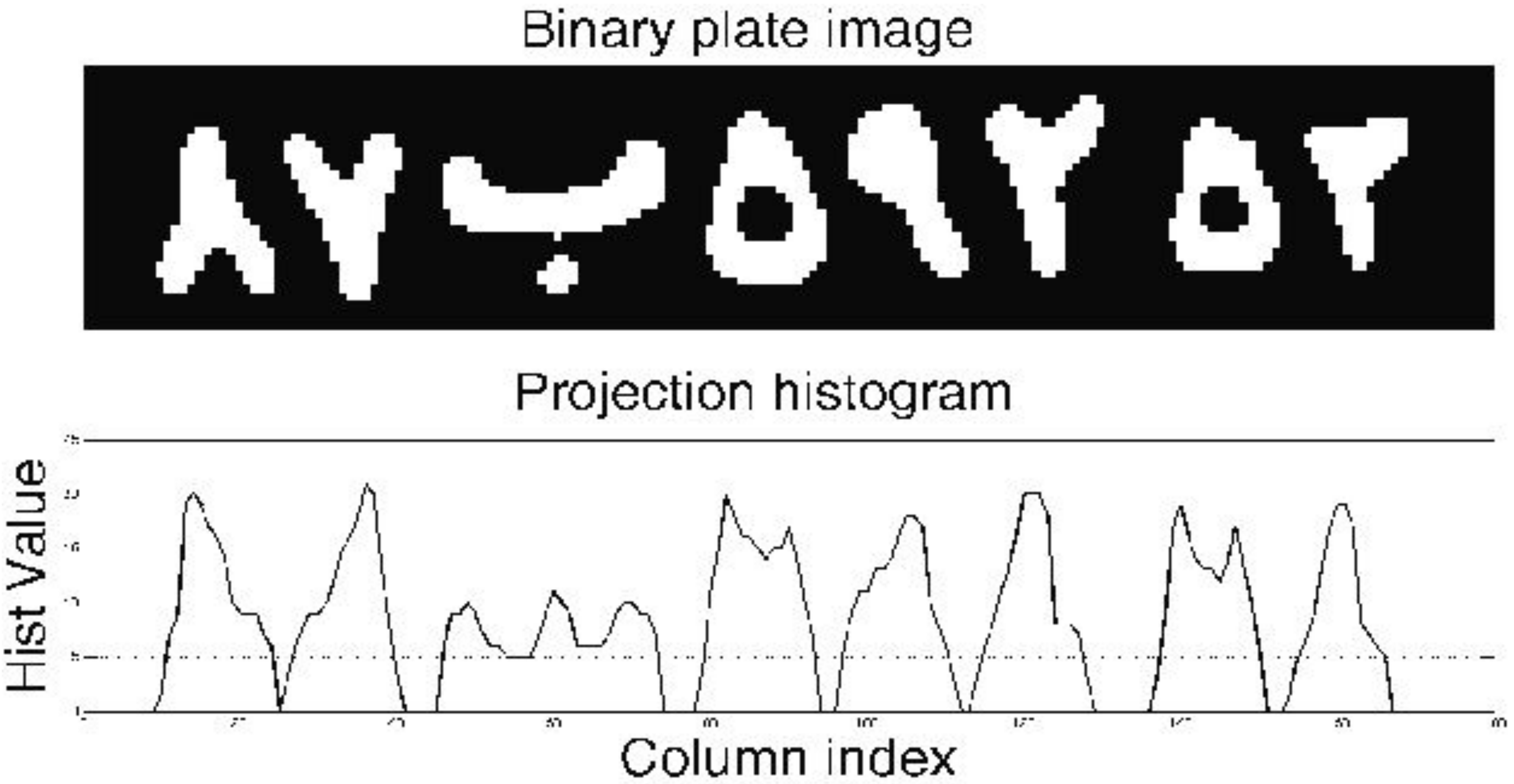
Histogram Equalization (Cont)





Histogram Projection (HP)

Assigns equal display space to every occupied raw signal level, regardless of how many pixels are at that same level. In effect, the raw signal histogram is "projected" into a similar-looking display histogram.


Binary plate image
Projection histogram
Hist Value
Column index

Histogram specification (HS)

an image's histogram is transformed according to a desired function

Transforming the intensity values so that the histogram of the output image approximately matches a specified histogram.

Histogram specification (Cont)