Convolutional neural networks (CNNs) pertama kali dikenalkan oleh Yann LeCun et all., pada tahun 1998 dalam makalahnya “Gradient-Based Learning Applied to Document Recognition”[[22]](http://yann.lecun.com/exdb/publis/pdf/lecun-01a.pdf). LeCun mengenalkan versi awal CNN yaitu LeNet (berasal dari nama LeCun) yang berhasil mengenali karakter tulisan tangan. Pada saat itu LeNet hanya mampu bekerja dengan baik pada gambar dengan resolusi rendah.

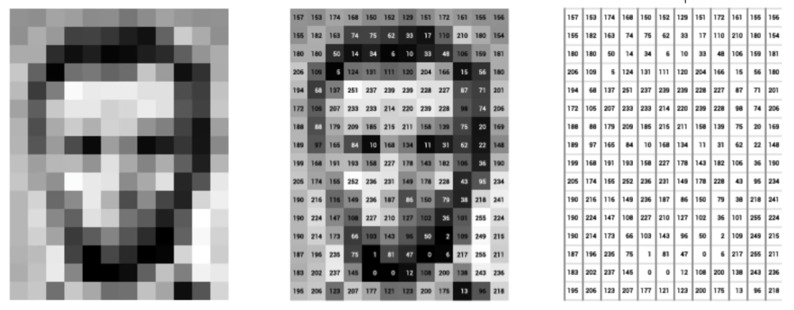
Database yang digunakan dalam LeCun adalah [MNIST database of handwritten digits](http://yann.lecun.com/exdb/mnist/index.html), terdiri dari pasangan angka 0 hingga 9 dengan labelnya. Dataset MNIST dikenal luas hingga saat ini dan banyak digunakan terutama oleh para pemula untuk melatih model machine learning.

Sejak ditemukannya LeNet, para peneliti terus melakukan riset untuk mengembangkan model CNN. Hingga pada tahun 2012, Alex Krizhevsky memperkenalkan AlexNet [[23]](https://papers.nips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf), versi lebih canggih dari CNN yang memenangkan perlombaan terkenal [ImageNet](http://image-net.org/index). AlexNet ini adalah cikal bakal deep learning, salah satu cabang AI yang menggunakan m*ulti-layer neural networks.*

Selain deep learning, salah satu bidang menarik yang muncul dari perkembangan machine learning adalah computer vision. Computer Vision adalah bidang yang memberi komputer kemampuan untuk ‘melihat’ seperti manusia.

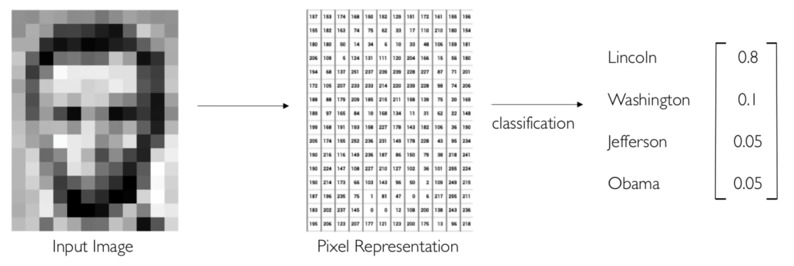
**Bagaimana Komputer Melihat**

Komputer adalah sebuah mesin yang hanya bisa membaca angka. Sebuah gambar dalam komputer adalah matriks yang berisi nilai dari setiap pixel di gambar. Pada gambar hitam putih, gambar merupakan matriks 2 dimensi. Pada gambar berwarna, gambar merupakan matriks 2 kali 3 dimensi di mana 3 dimensi terakhir adalah jumlah kanal dari gambar berwarna yaitu **RGB**(red, green, blue).

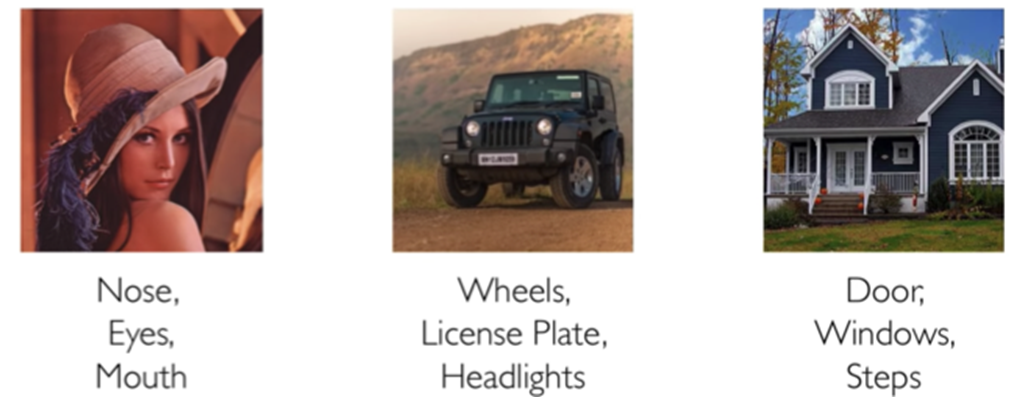
[](https://www.dicoding.com/academies/184/tutorials/8512?from=8507)

**Klasifikasi Gambar**

Salah satu peran machine learning dalam computer vision adalah pada klasifikasi gambar. Contohnya, kita punya label yaitu nama beberapa presiden Amerika Serikat. Kita ingin memprediksi siapa presiden di gambar. Jaringan saraf seperti di bawah menunjukkan probabilitas siapa presiden di dalam foto.

[](https://www.dicoding.com/academies/184/tutorials/8512?from=8507)

Untuk mengklasifikasikan siapa presiden dalam gambar dengan benar, model kita harus mampu untuk mengenali ciri-ciri unik yang terdapat pada wajah Lincoln, Washington, Jefferson, dan Obama.

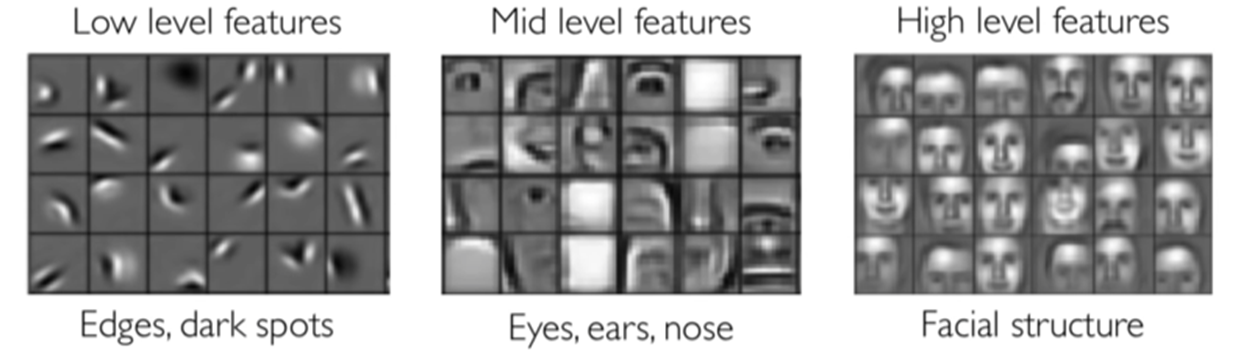
[](https://www.dicoding.com/academies/184/tutorials/8512?from=8507)

Setiap objek dari gambar memiliki atribut unik. Seperti pada gambar di atas, objek orang memiliki atribut unik berupa hidung, mata, dan mulut. Pada objek mobil terdapat atribut roda, lampu, dan plat nomor, sedangkan objek rumah memiliki atribut pintu, jendela, dan tangga.

**Convolutional Layer**

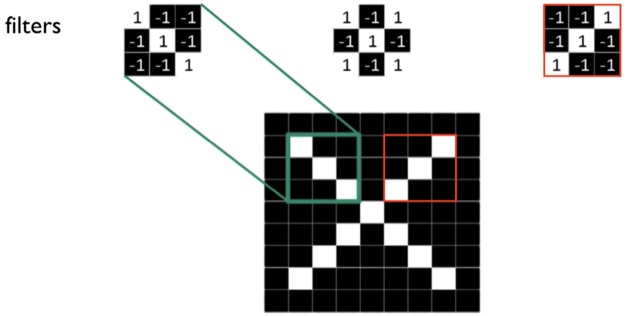
Sebuah jaringan saraf biasa mengenali gambar berdasarkan piksel-piksel yang terdapat pada gambar. Teknik yang lebih optimal adalah dengan menggunakan convolutional layer di mana alih alih mengenali objek berdasarkan piksel-piksel, jaringan saraf dapat mengenali objek berdasarkan atribut-atribut yang memiliki lebih banyak informasi.

Convolutional layer berfungsi untuk mengenali atribut-atribut unik pada sebuah objek.  Atribut-atribut yang lebih rendah membentuk atribut lebih tinggi contohnya atribut wajah dibentuk dari atribut mata, telinga, dan hidung. Atribut mata dibentuk dari garis, lengkungan dan bintik hitam.

[](https://www.dicoding.com/academies/184/tutorials/8512?from=8507)

**Filter**

Convolutional layer dapat mengenali atribut pada objek menggunakan filter. Filter hanyalah sebuah matriks yang berisi angka-angka. Apa maksudnya? Perhatikanlah gambar di bawah ini.

[](https://www.dicoding.com/academies/184/tutorials/8512?from=8507)

Pada gambar tersebut, terdapat 3 buah filter yang merupakan matriks 3x3 dan sebuah objek gambar berupa huruf X. Filter yang berada di sebelah kiri digambarkan dapat mengenali garis yang terdapat pada kotak hijau. Setiap filter berbeda dapat mengenali atribut yang berbeda seperti, filter di kanan dapat mengenali atribut objek x yang berada di kotak merah.

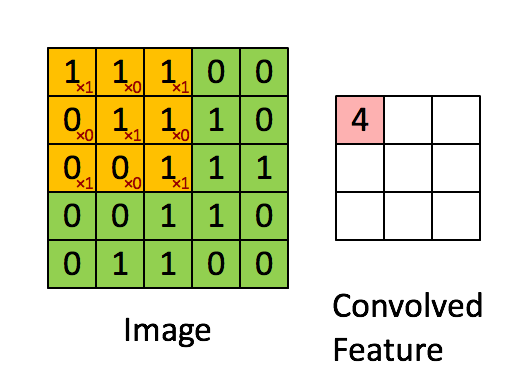
Contoh lain dari filter dapat Anda lihat di bawah. Pada sebuah gambar perempuan, aplikasi dari filter yang berbeda menghasilkan gambar yang berbeda.

Kita dapat membedakan seekor kuda dan manusia berdasarkan bentuknya bukan? Nah, dengan filter seperti pada gambar yang paling kanan, kita dapat mendeteksi garis-garis yang bisa menunjukkan apakah seseorang merupakan kuda atau manusia berdasarkan bentuk garisnya.

[](https://www.dicoding.com/academies/184/tutorials/8512?from=8507)

**Proses Konvolusi**

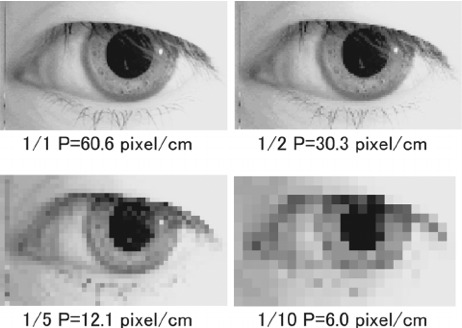
Proses konvolusi adalah proses yang mengaplikasikan filter pada gambar. Pada proses konvolusi ada perkalian matriks terhadap filter dan area pada gambar. Pada ilustrasi di bawah terdapat sebuah gambar, filter, dan hasil dari proses konvolusi terhadap gambar. Animasi selanjutnya menunjukkan bagaimana proses konvolusi pada ilustrasi sebelumnya dikerjakan.

[](https://www.dicoding.com/academies/184/tutorials/8512?from=8507)

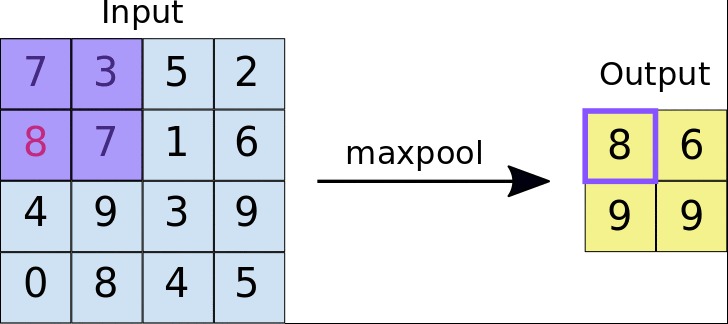
Ketika proses konvolusi selesai, hasil dari konvolusi tersebut dapat dijadikan masukan untuk sebuah MLP.

**Max Pooling**

Pada sebuah jaringan saraf tiruan, umumnya setelah proses konvolusi pada gambar masukan, akan dilakukan proses pooling. Pooling adalah proses untuk mengurangi resolusi gambar dengan tetap mempertahankan informasi pada gambar. Contohnya seperti pada gambar berikut di mana ketika resolusi dikurangi sampai batas tertentu kita masih bisa mendapatkan informasi mengenai objek pada gambar.

[](https://www.dicoding.com/academies/184/tutorials/8512?from=8507)

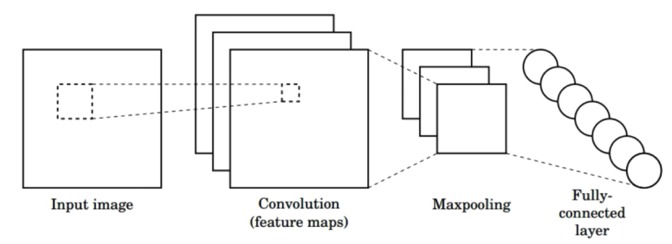
Salah satu contoh dari pooling adalah max pooling. Pada max pooling di antara setiap area dengan luas piksel tertentu, akan diambil satu buah piksel dengan nilai tertinggi. Hasilnya akan menjadi gambar baru. Animasi di bawah menunjukkan contoh max pooling dengan ukuran 2x2 piksel pada gambar berukuran 4x4 piksel. Hasil dari max pooling adalah gambar dengan ukuran 2x2 piksel.

[](https://www.dicoding.com/academies/184/tutorials/8512?from=8507)

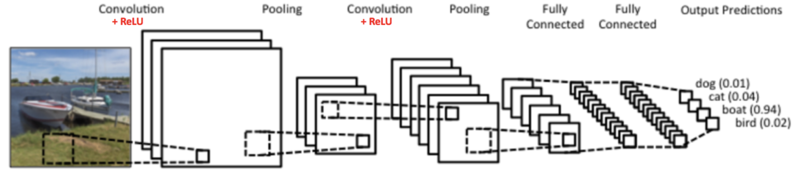
Proses max pooling dipakai karena pada praktiknya, jumlah filter yang digunakan pada proses konvolusi berjumlah banyak. Ketika kita menggunakan 64 filter pada konvolusi maka akan menghasilkan 64 gambar baru. Max pooling membantu mengurangi ukuran dari setiap gambar dari proses konvolusi.

**Arsitektur Convolutional Neural Network**

Arsitektur CNN adalah sebuah jaringan saraf yang menggunakan sebuah layar konvolusi dan max pooling. Pada arsitektur CNN di bawah, sebuah gambar masukan dideteksi atribut/fitur nya dengan menggunakan konvolusi 3 filter. Lalu setelah proses konvolusi akan dilakukan max pooling yang menghasilkan 3 buah gambar hasil konvolusi yang memiliki resolusi lebih kecil. Terakhir, hasil max pooling dapat dimasukkan ke dalam sebuah hidden layer MLP.

[](https://www.dicoding.com/academies/184/tutorials/8512?from=8507)

Kita juga dapat menggunakan beberapa lapis konvolusi dan max pooling sebelum mulai memasukkannya ke hidden layer sebuah MLP. Cara kerjanya sederhana. Kita bisa melakukan proses konvolusi dan max pooling setelah lapisan max pooling sebelumnya. Pada contoh di bawah terdapat 2 kali proses konvolusi dan max pooling sebelum hasilnya dimasukkan ke dalam hidden layer.

[](https://www.dicoding.com/academies/184/tutorials/8512?from=8507)

Dengan beberapa lapis proses konvolusi, makin detail fitur yang dapat dikenali dari gambar. Contohnya pada proses konvolusi pertama dapat mendeteksi wajah dari seorang manusia. Lalu pada proses konvolusi kedua, wajah hasil konvolusi pertama dapat dideteksi fitur yang lebih detail seperti hidung, mata, dan telinganya sehingga, model makin pintar membedakan wajah setiap orang. Jika Anda ingin mengetahui lebih detail mengenai CNN, Anda dapat mengunjungi [tautan](https://cs231n.github.io/convolutional-networks/) berikut yah.

Selamat! Sekarang Anda paham cara kerja CNN dan fungsinya dalam deteksi gambar. Di modul berikutnya kita akan praktik mengembangkan sebuah model CNN dengan TensorFlow.