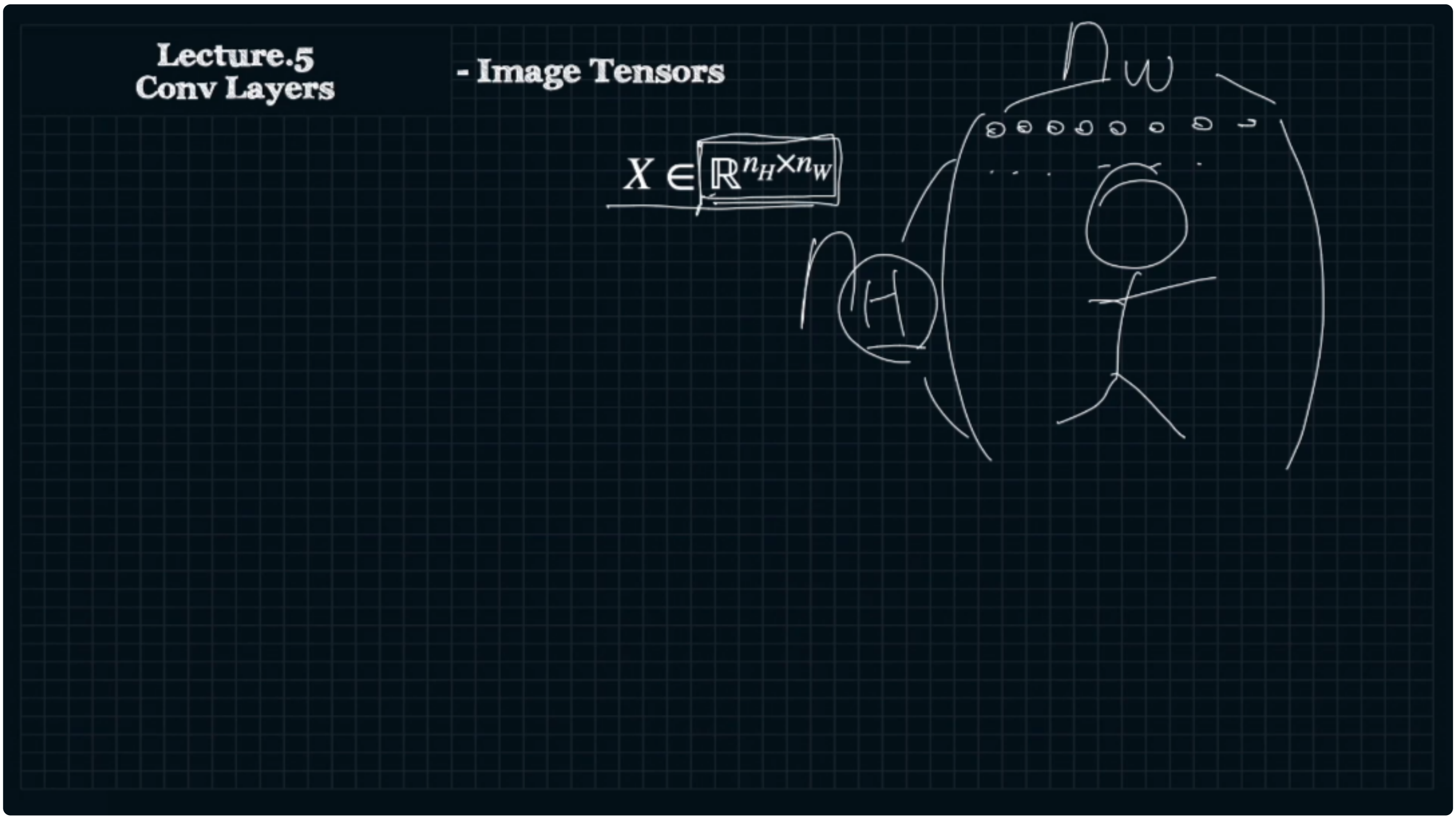
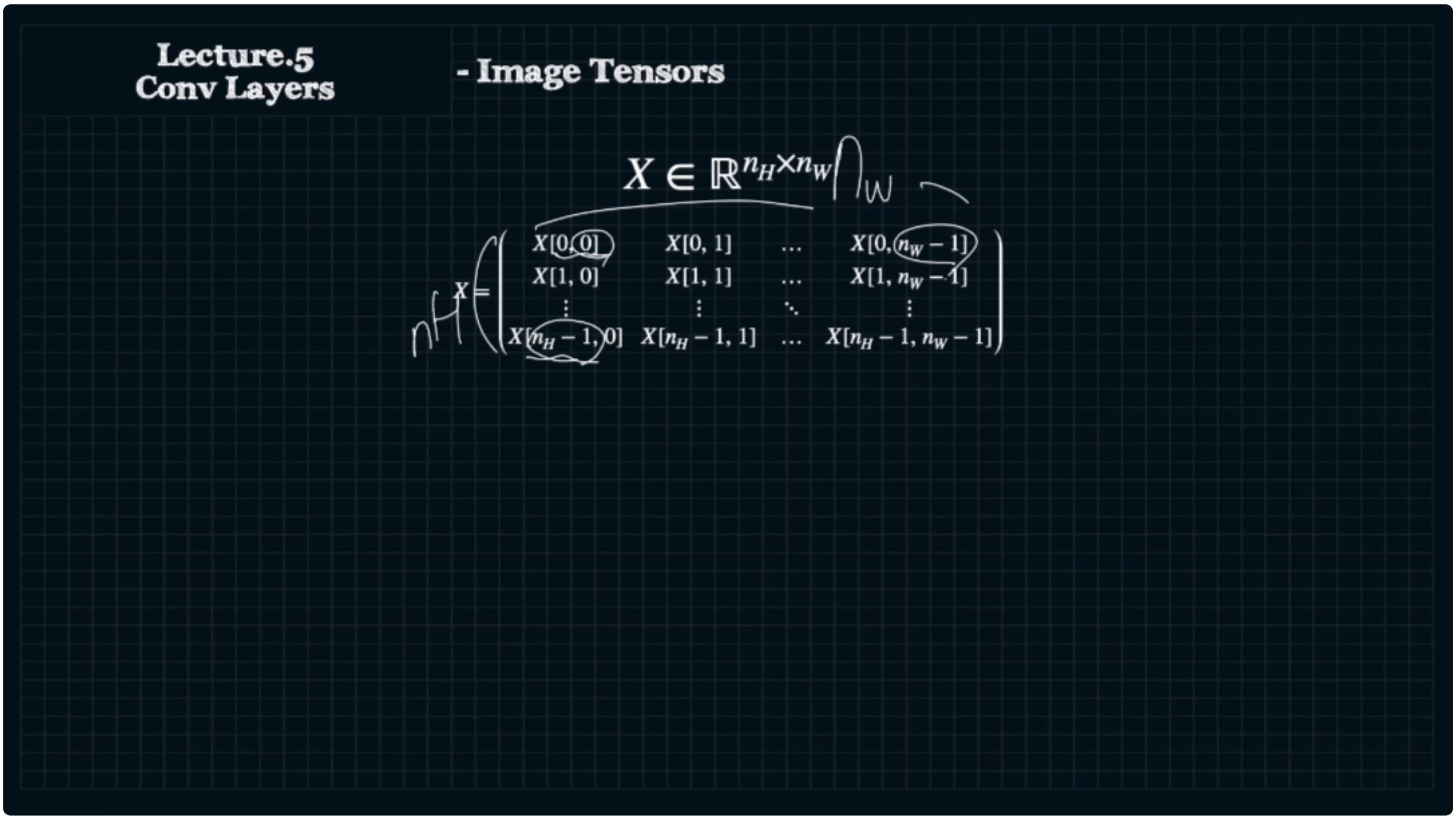
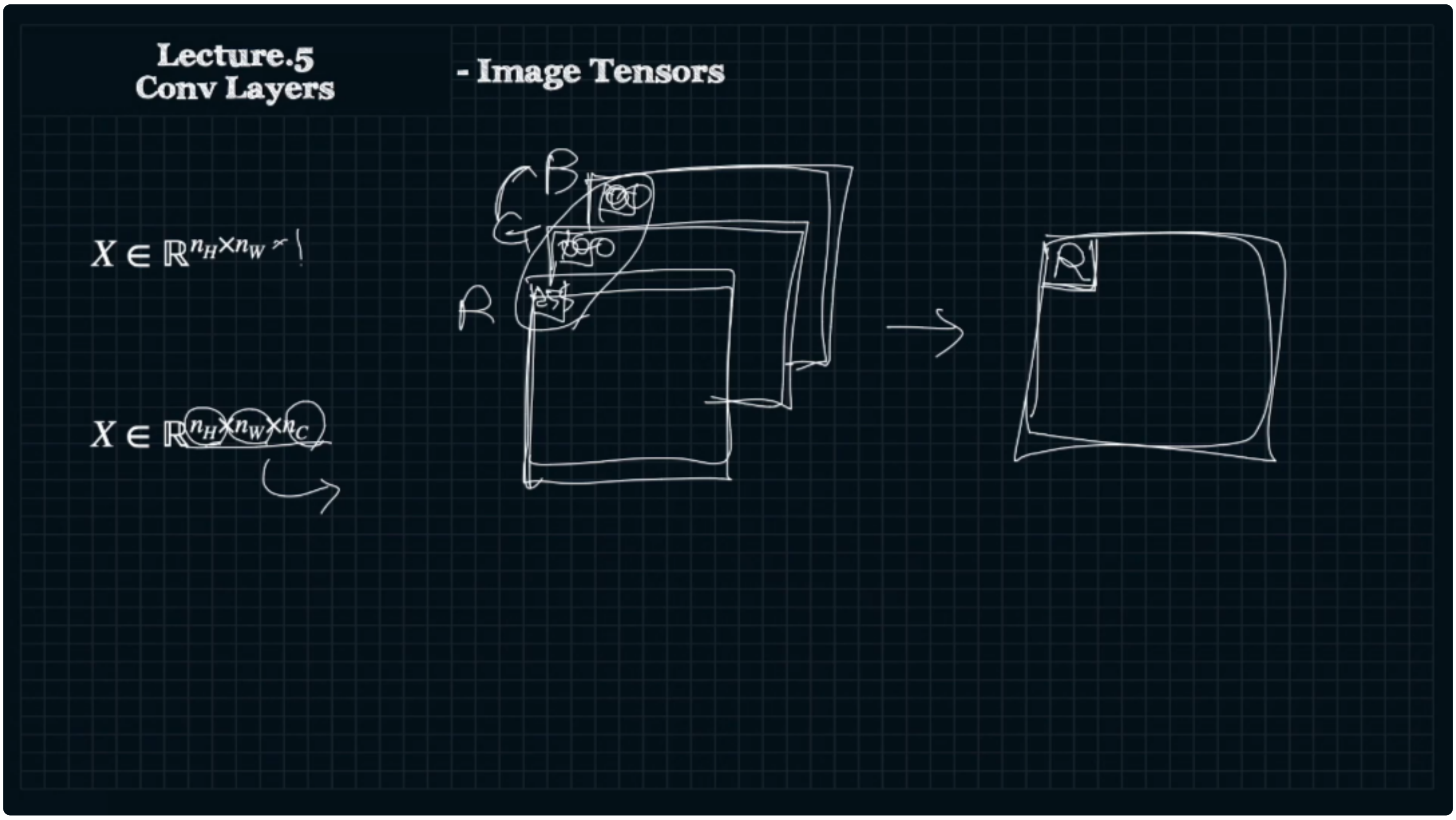


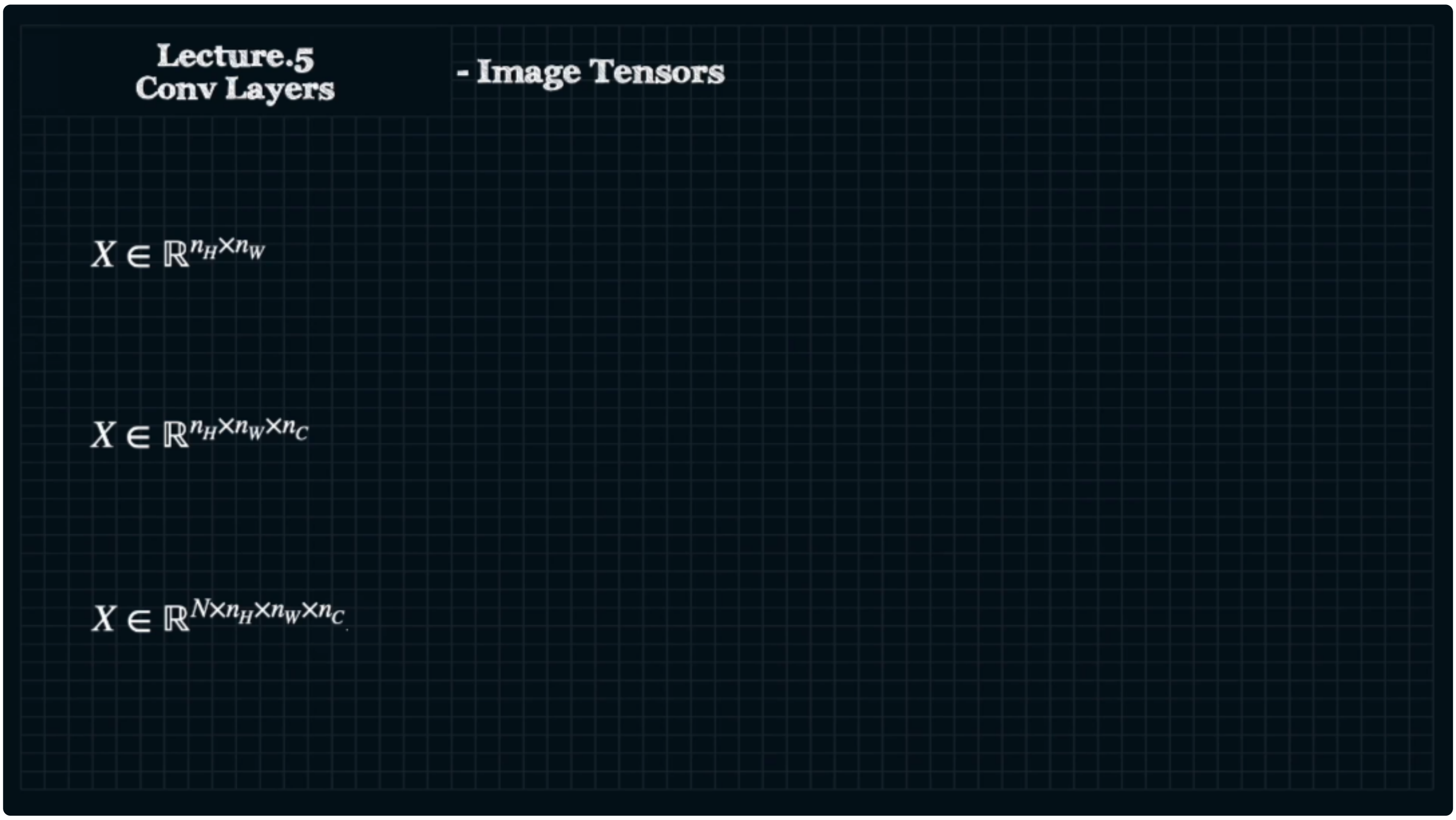
- 이미지를 행렬의 형태로 다룰 수 있다. 동그라미 하나하나가 픽셀이다.



- 행은 N_h만큼 표현이되고, 컬럼은 X_w만큼 표현이된다.

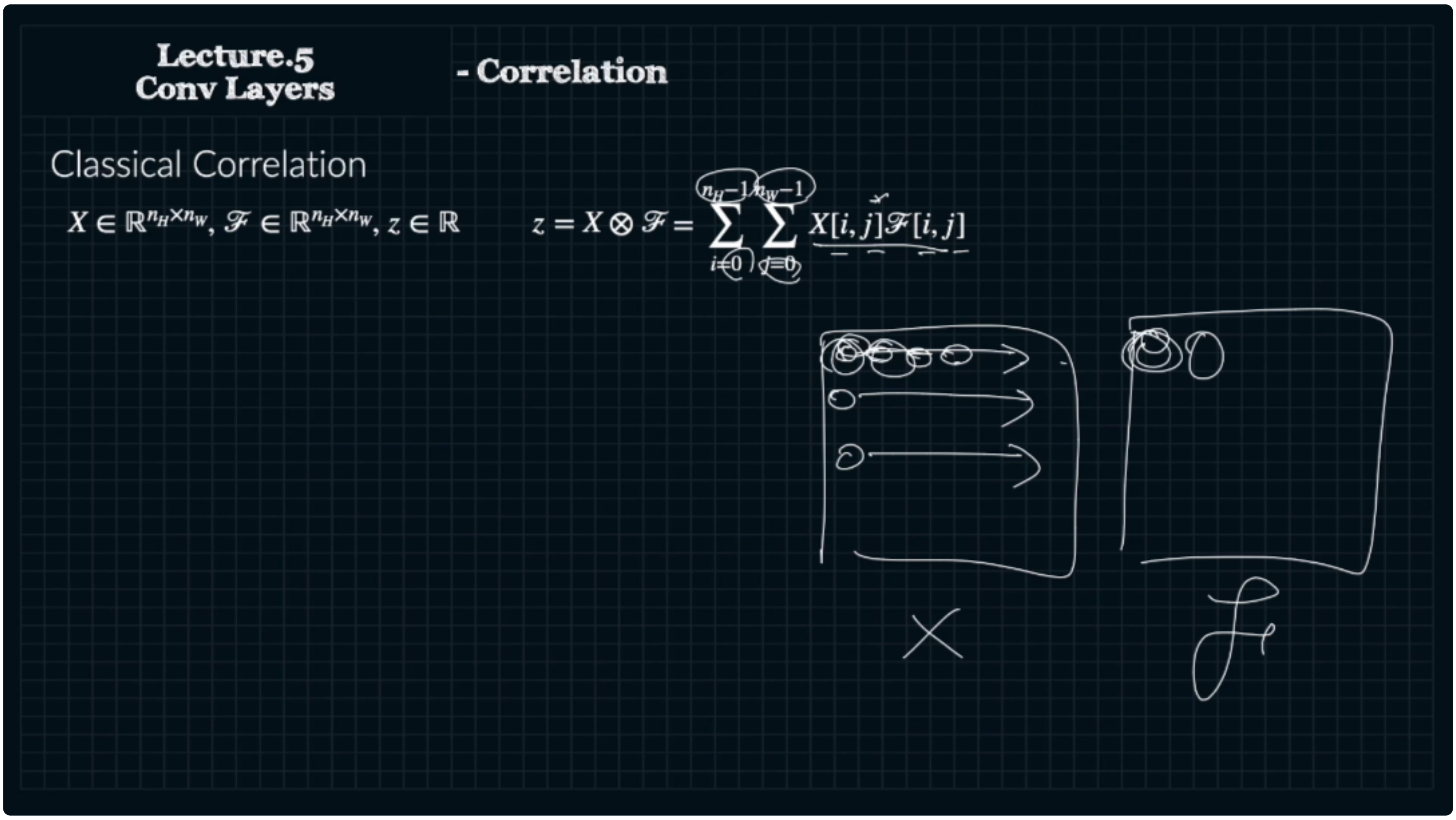


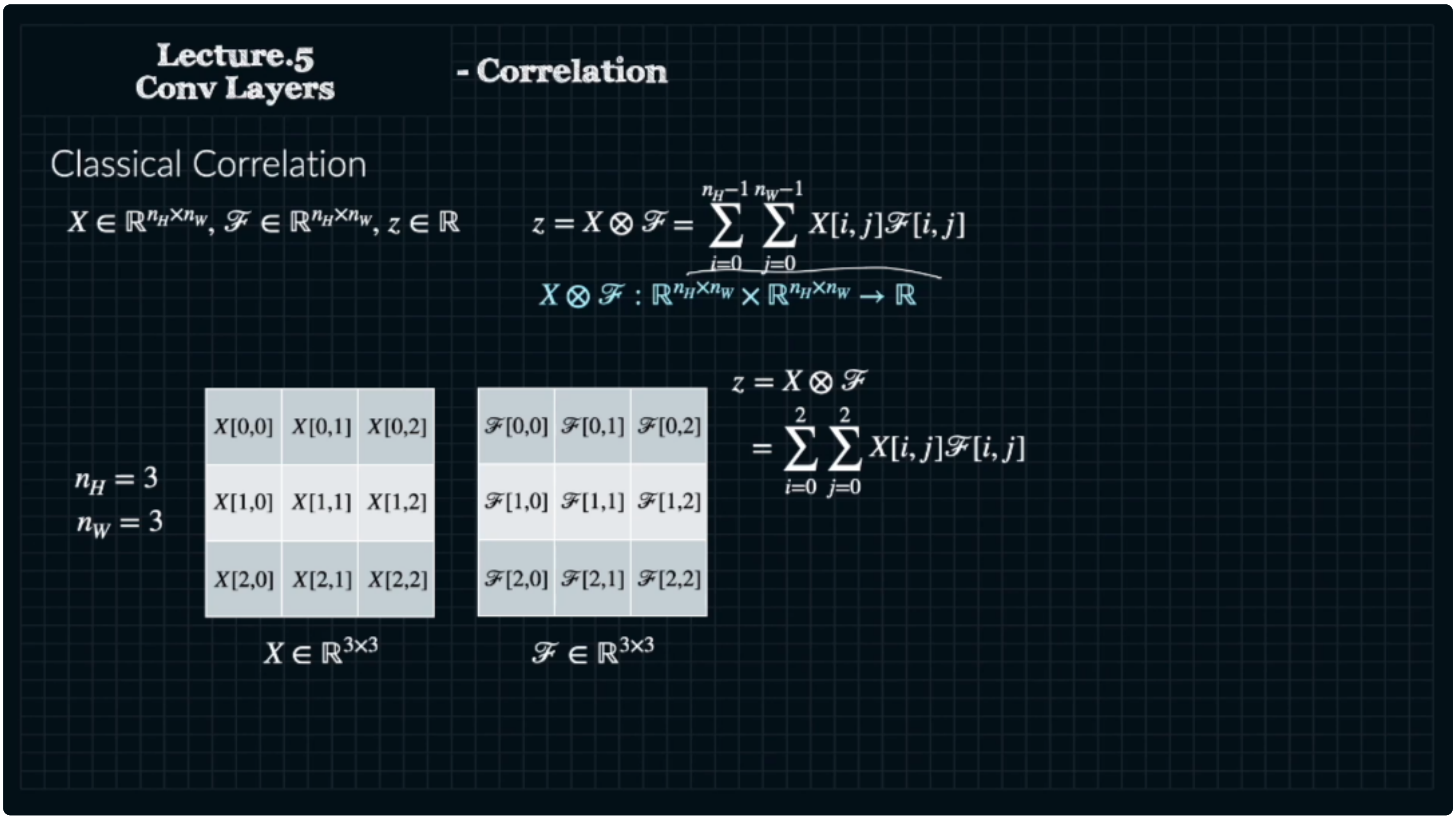
- 이미지를 RGB로 표현할 수 있는데 사진처럼 R(red) G(green) B(blue) 픽셀값을 가지고 있는 종이가 있다고 생각하면 된다. 그래서 3개의 종이를 합치면, 3차원이된다. 다시말하자면, R,G,B각각 metrix라고 생각하면된다.
- Height x Width x channels 이렇게 표현이된다.

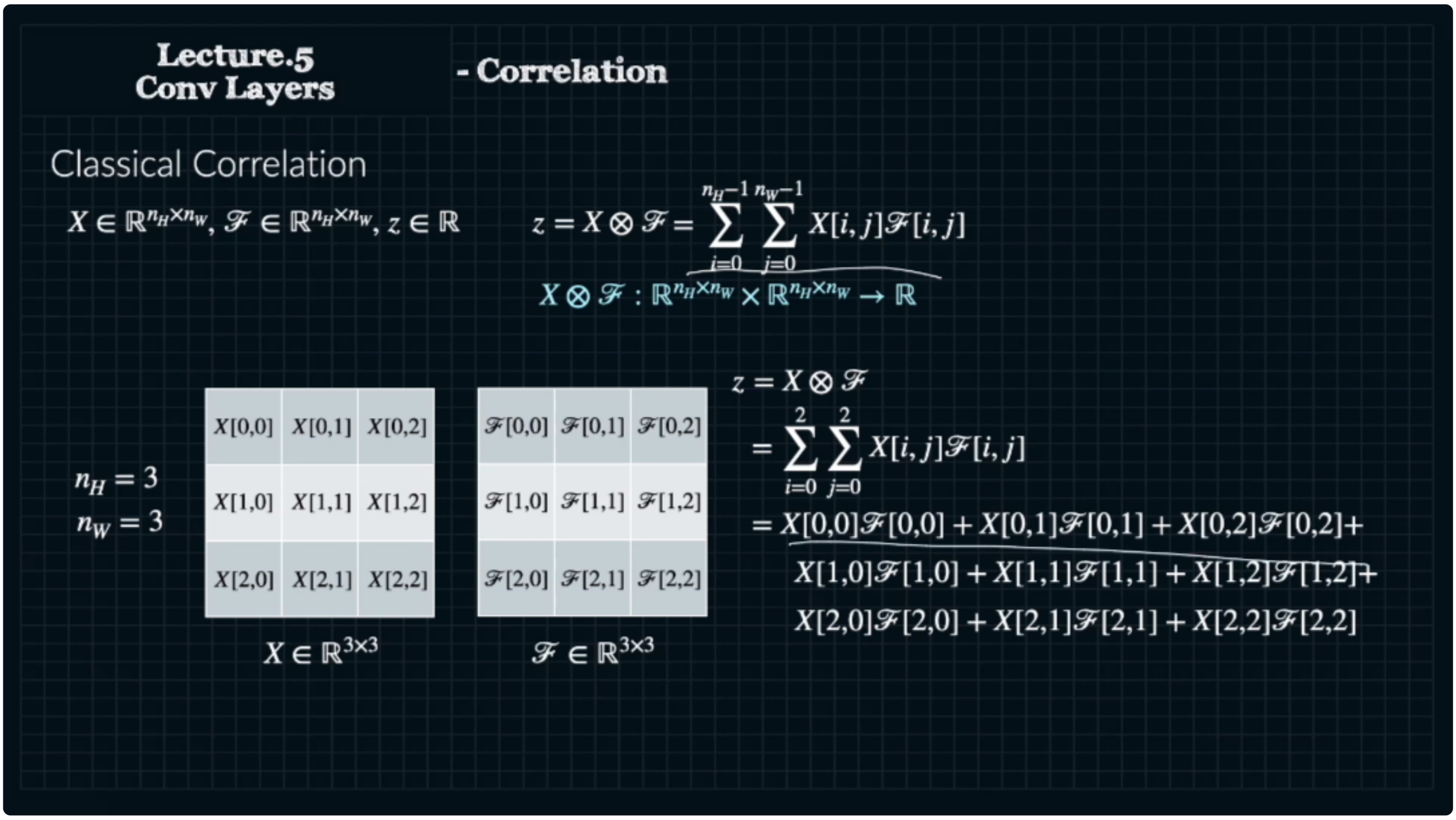


- color image가 batch_size만큼 있다고 표현 방법
- N x Height x Width x channels

우리가 아는 Convolution 내에서 연산은 모두 Correlation연산이다.

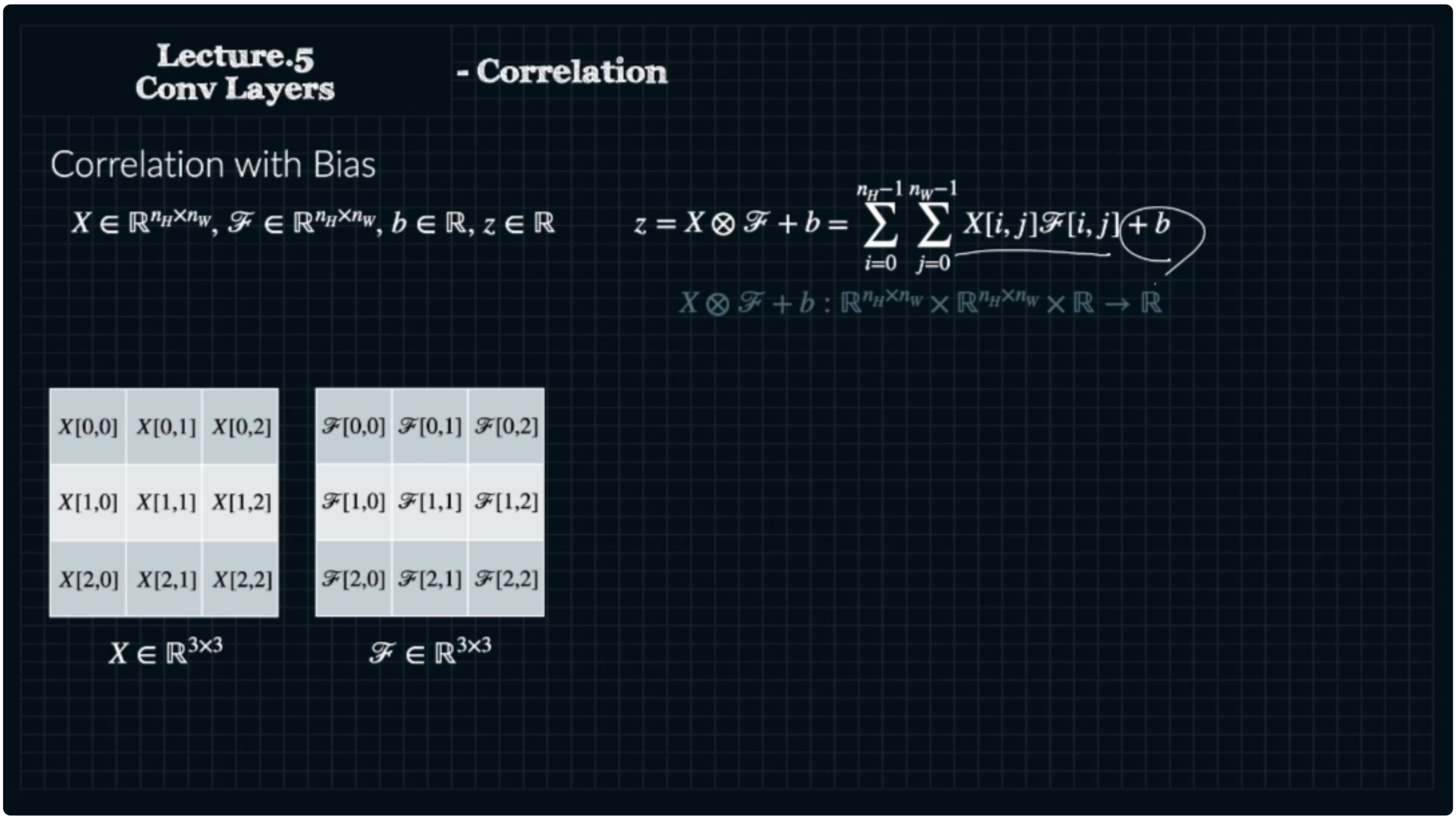






Correlation연산이 끝나면 하나의 scalar 값이 나온다. Correlation연산의 예를들어보자. 하나의 이미지와 하나의 필터가 있다고 하자. 이 이미지와 필터의 사이즈는 같다. 그래서 연산을 해보자면, x[0,0] x f[0,0] + x[0, 1] x f[0, 1] ..... 이렇게 행렬의 왼쪽 위부터 아래오른쪽 모서리까지 연산이된다.

사실 Corrleation은 신호처리에서 이 x와 f가 얼마나 닮아있는지를 확인하기 위한 연산이다.



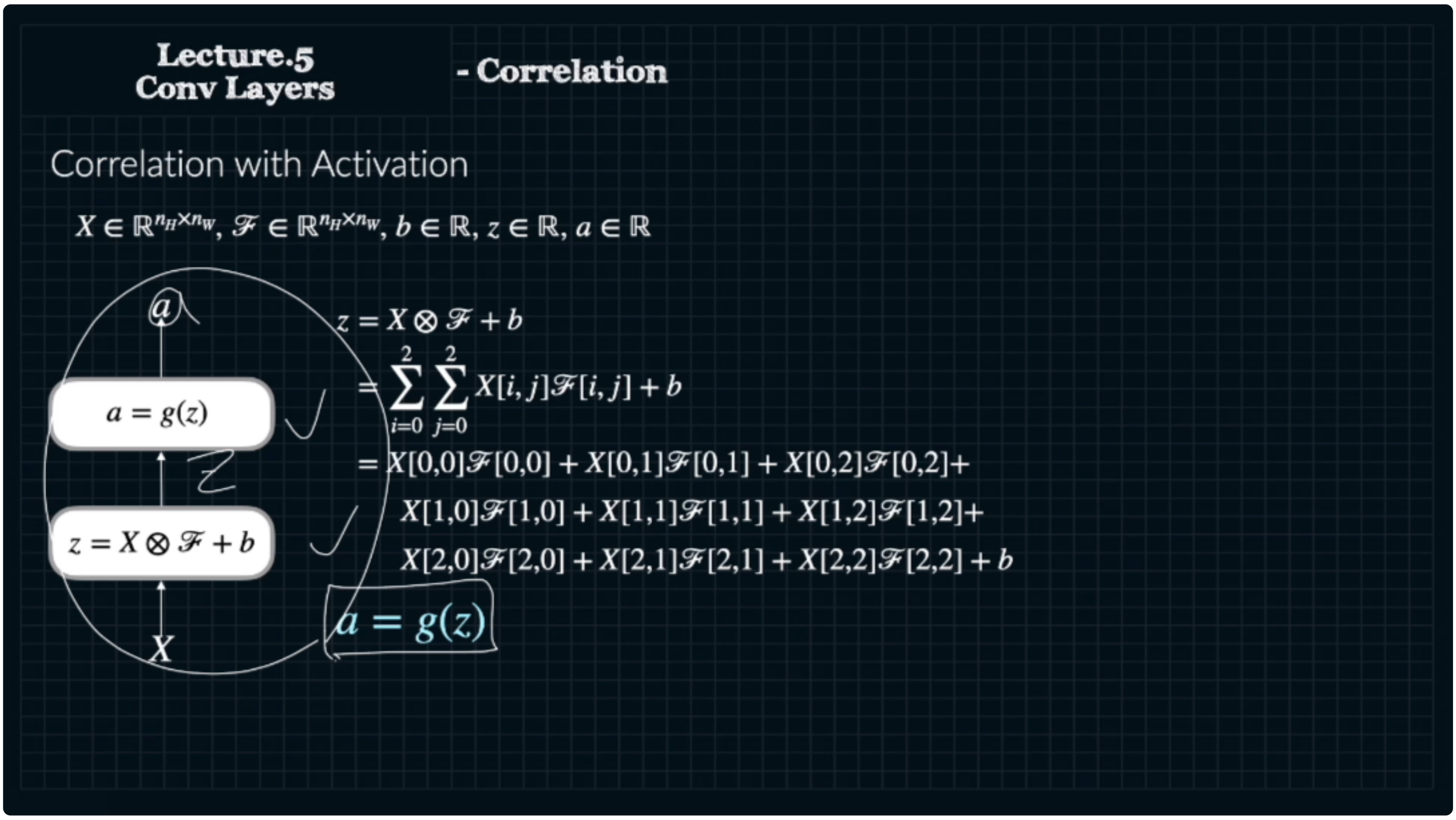
z값을 연산하기위해서 bias가 포함이된다. 앞에가 weight가 되고 뒤에가 bias가 된다.

아까 앞에서 했던 correlation연산과 같으면서 뒤에 bias를 더해주면 된다. 이것이 딥러닝에서의 correlation 연산이된다.

그러면 Convolution연산은 어떻게하냐면, 필터를 반대로 뒤집은다음에 이미지와 연산을 하면된다.

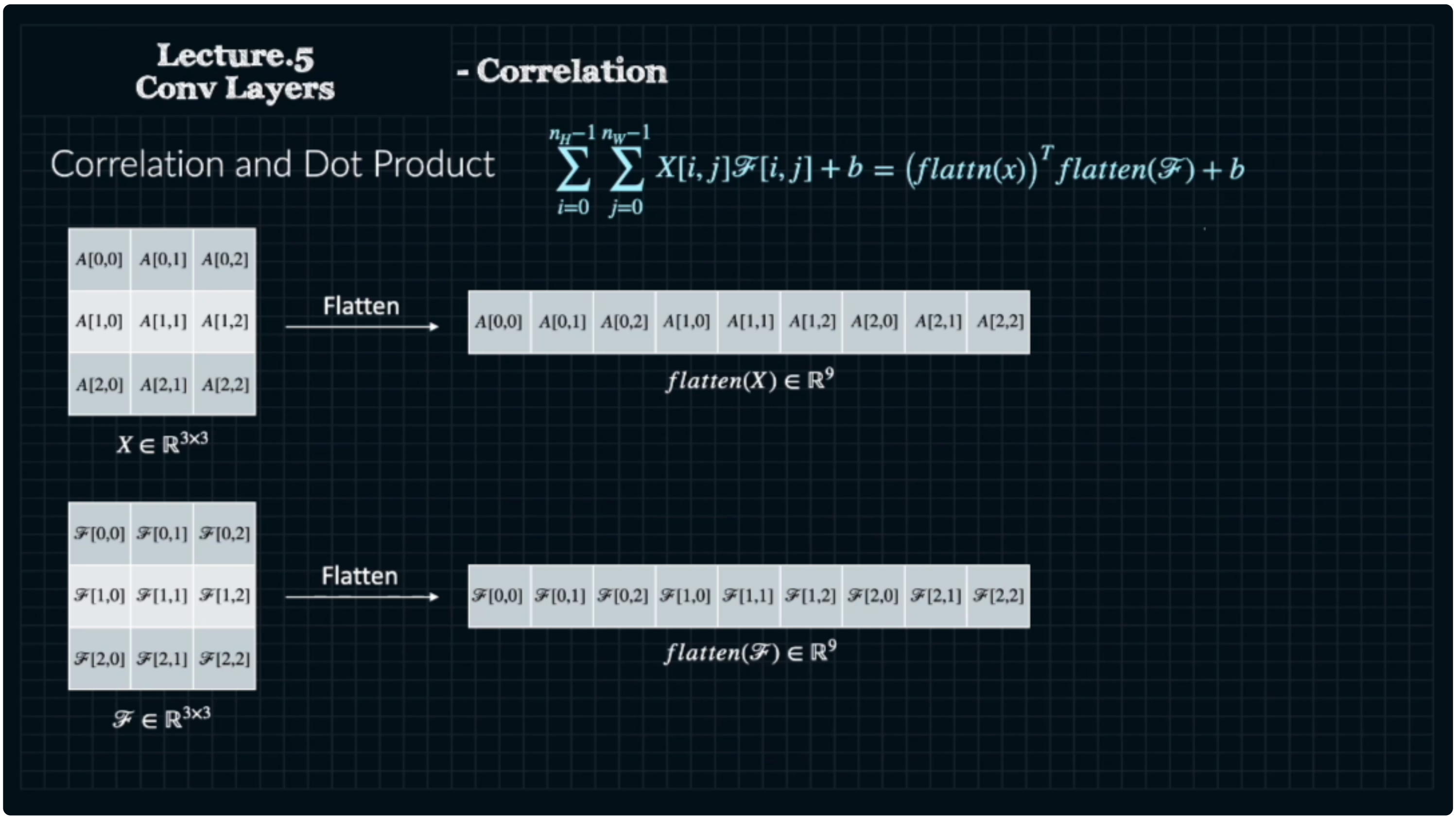
신호처리에서의 correlation은 x와 필터(두 신호)가 얼마나 비슷한지에 대한 값을 뽑는 걸 생각하면되고, 딥러닝에서 convolution은 필터를 통과한 아웃풋을 계산해주는 도구라고 생각하면된다.

우리가 아는 convolution은 알고보면 신호처리의 correlation 연산이고 correlation연산은 알고보면 두 신호가 얼마나 비슷하지에 대한 값을 뽑는 것이라고 생각하면된다.



Correlation에서 나온값이 우리가 이전부터 해오던 Affine function에서 나온 값인 z값이라고 생각하면된다.

Affine function을 통과하면 activation을 통과하게 된다.



Correlation연산은 알고보면 Dot product 연산이랑 다른게 없다.

행렬을 Flatten을 시켜주면 일차원인 vector형태로 펼쳐지는데 여기서 x를 Transformation시켜주고 필터와 연산하고 여기에 bias을 더해주면 Affine transformation연산이되는 것이다.