

Understand and Build CNN from the Ground Up and Intuition

Nhóm AIO_TimeSeries

Ngày 18 tháng 11 năm 2025

Bài viết này giới thiệu về Convolutional Neural Networks (CNN) và cách nó hoạt động bằng cách (filter perspective) thay vì cách nhìn bằng kernel. Mình sẽ giải thích cách CNN truyền thông tin qua các lớp CNN "nhỏ" này, cách lan truyền thông tin qua các khía cạnh vị trí của chúng cao hơn so với 1x1 Convolution.

Phn 1: Ti sao Neural Network truyn thng (MLP) "bú tay" vi hơnh nh?

Phn 2: T duy B lc (Filter Perspective) & C ch c bn

Phn 3: Kin trúc CNN - Xp tng cùc khi x lý

Phn 4: CNN Backpropagation & S tht v phđp xoay Kernel

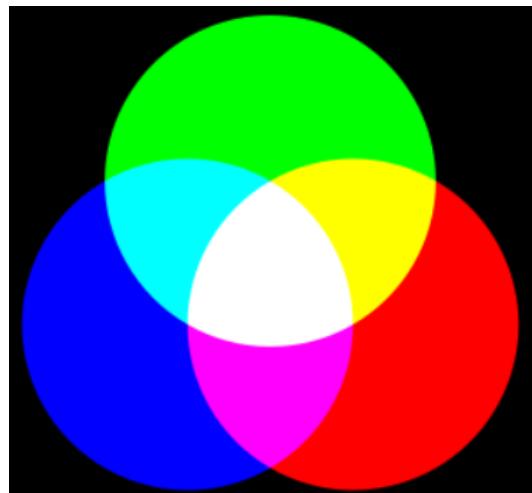
Phn 5: CNN Nóng cao - 1x1 Convolution & Phón bit chiu khñg gian

Phn 1: Ti sao Neural Network truyn thng (MLP) "bú tay" vi hơnh nh?

Trong khi máy tính có thể học hỏi từ dữ liệu, tại sao mô hình không thể làm điều đó mà chỉ có thể thực hiện các phép tính tuyến性 (MLP) thô sơ? Điều này là do khái niệm "máy tính" không có khả năng nhận biết hình ảnh như con người.

1.1 Bn cht ca mịu sc (RGB Color)

Trong thời gian đầu, máy tính chỉ có thể nhận biết màu bằng cách phân tích các khung ảnh thành các khung ảnh nhỏ hơn, sau đó áp dụng các thuật toán để xác định màu sắc. Tuy nhiên, cách tiếp cận này không hiệu quả và tốn nhiều thời gian. Sau đó, kỹ thuật CNN đã được phát triển, cho phép máy tính nhận biết hình ảnh một cách tự động và nhanh chóng.



1.2 Biểu diễn dữ liệu Ma trận (Matrix Representation)

Một bức ảnh có thể được xem là một mảng dữ liệu với kích thước 800×600 và có 3 kênh màu (Red, Green, Blue).

$$\begin{bmatrix} w_{1,1} & w_{1,2} & \dots & w_{1,800} \\ w_{2,1} & w_{2,2} & \dots & w_{2,800} \\ \dots & \dots & \dots & \dots \\ w_{600,1} & w_{600,2} & \dots & w_{600,800} \end{bmatrix} \quad \begin{bmatrix} (100, 100, 50) & (101, 112, 3) & (131, 20, 80) \\ (150, 210, 130) & (10, 120, 130) & (111, 120, 130) \\ (10, 260, 30) & (200, 20, 30) & (100, 20, 3) \end{bmatrix}$$

(a) Ma trn tng quót $W_{i,j}$

(b) Giò tr pixel lị b 3 s (R,G,B)

Ti mi v trờ ta (i, j) , giò tr pixel w_{ij} khũng tn ti di dng mt s n l mị lị mt b ba giò tr (r_{ij}, g_{ij}, b_{ij}) , vờ d nh $(0, 233, 256)$ to ra mịu xanh ngc. ti u húa cho vic lu tr vị x lỳ tòn thng khũng gp chung mị tòn riồng cùc giò tr niy thịnh 3 ma trn c lp tng ng vi 3 kǒnh mị , Xanh lò vị Xanh dng.

$$\begin{bmatrix} 100 & 101 & 131 \\ 150 & 10 & 111 \\ 10 & 200 & 100 \end{bmatrix}, \begin{bmatrix} 100 & 112 & 20 \\ 210 & 120 & 120 \\ 260 & 20 & 20 \end{bmatrix}, \begin{bmatrix} 50 & 3 & 80 \\ 130 & 130 & 130 \\ 30 & 30 & 3 \end{bmatrix}$$

R **G** **B**

(c) Tònch mt nh mịu thịnh 3 ma trn R, G, B riồng bit

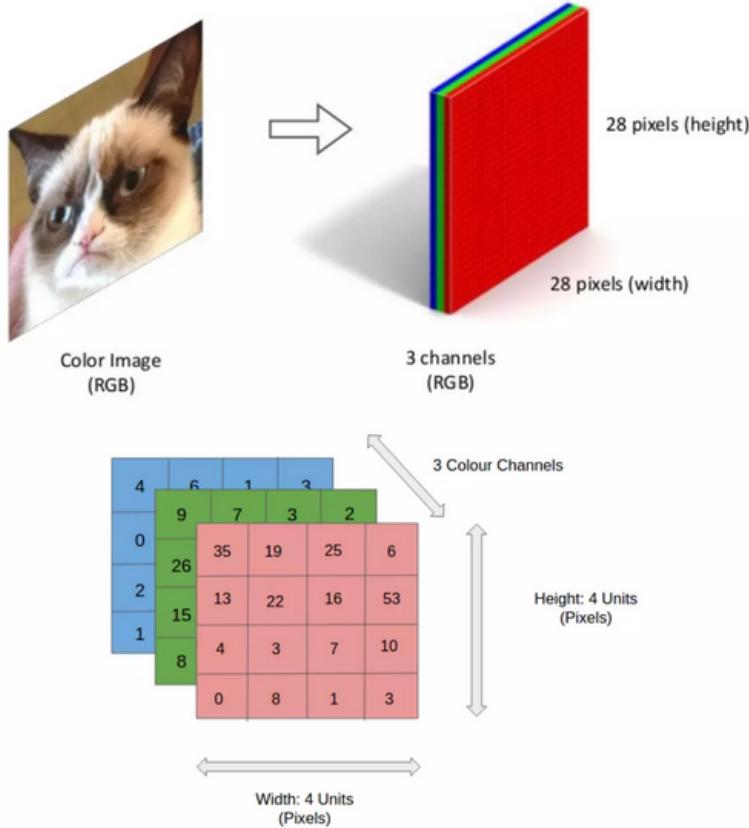
$$\begin{bmatrix} (r_{1,1}, g_{1,1}, b_{1,1}) & (r_{1,2}, g_{1,2}, b_{1,2}) & \dots & (r_{1,800}, g_{1,800}, b_{1,800}) \\ (r_{2,1}, g_{2,1}, b_{2,1}) & (r_{2,2}, g_{2,2}, b_{2,2}) & \dots & (r_{2,800}, g_{2,800}, b_{2,800}) \\ \dots & \dots & \dots & \dots \\ (r_{600,1}, g_{600,1}, b_{600,1}) & (r_{600,2}, g_{600,2}, b_{600,2}) & \dots & (r_{600,800}, g_{600,800}, b_{600,800}) \end{bmatrix}$$

$$\Rightarrow \begin{bmatrix} r_{1,1} & r_{1,2} & \dots & r_{1,800} \\ r_{2,1} & r_{2,2} & \dots & r_{2,800} \\ \dots & \dots & \dots & \dots \\ r_{600,1} & r_{600,2} & \dots & r_{600,800} \end{bmatrix}, \begin{bmatrix} g_{1,1} & g_{1,2} & \dots & g_{1,800} \\ g_{2,1} & g_{2,2} & \dots & g_{2,800} \\ \dots & \dots & \dots & \dots \\ g_{600,1} & g_{600,2} & \dots & g_{600,800} \end{bmatrix}, \begin{bmatrix} b_{1,1} & b_{1,2} & \dots & b_{1,800} \\ b_{2,1} & b_{2,2} & \dots & b_{2,800} \\ \dots & \dots & \dots & \dots \\ b_{600,1} & b_{600,2} & \dots & b_{600,800} \end{bmatrix},$$

1.3 T Ma trn n Tensor

Khòi nim **Tensor** xut hin tng quót húa cùc cu truct d liu niy theo s chiu. Trong khi Vector lị Tensor 1D vị Ma trn lị Tensor 2D (nh nh xòm ch cn mt ma trn duy nht biu din sòng t en sang trng), tho nh mịu li phc tp hn. Võ nh mịu c to thịnh t vic chng 3 ma trn R, G, B lồn nhau (collapse on top of each other), nú c nh ngha lị mt **Tensor 3D** vi kờch thc *Height* \times *Width* \times *Depth* (trong ú *Depth* = 3).

color image is 3rd-order tensor



Vic hiu ũng chiu sóu (Depth) ca Tensor lị mu cht hiu còch CNN vn hịnh sau niy.

1.4 S bứt n tham s (The Parameter Explosion)

Chờnh cu truct Tensor 3D niy lị nguyễn nhón khin mng MLP truyn thng "u hinct". Ch th lịm mt phôp tònñ vi bc nh rt nh körch thec 64×64 : tng s giò tr u vio s lị $64 \times 64 \times 3 = 12,288$. Nu ta kt ni u vio niy vio mt lp n ch gm 1,000 n-ron theo còch kt ni tojn b (fully connected) ca MLP, s lñg trng s cn hc s lñn ti hn **12 triu** tham s ($12,288 \times 1,000$). Khi lñg tònñ toòn khng l niy khñng ch lịm chm h thng mi cùn dn n hin tng Overfitting nghiòm trng, buc mñnh phi tóm n gii phòp thñng minh hn lị CNN x lỳ cu truct Tensor mt còch hiu qu.

Phn 2: T duy B lc (Filter Perspective) & C ch c bn

óy lị phn trng tóm hiu CNN t gc r. Thay vø t duy theo kiu "kt ni tt c mi th" nh mng MLP truyn thng, CNN tip cn hñnh nh theo còch con ngi quan sòt: tóm kim còc c trng (features).

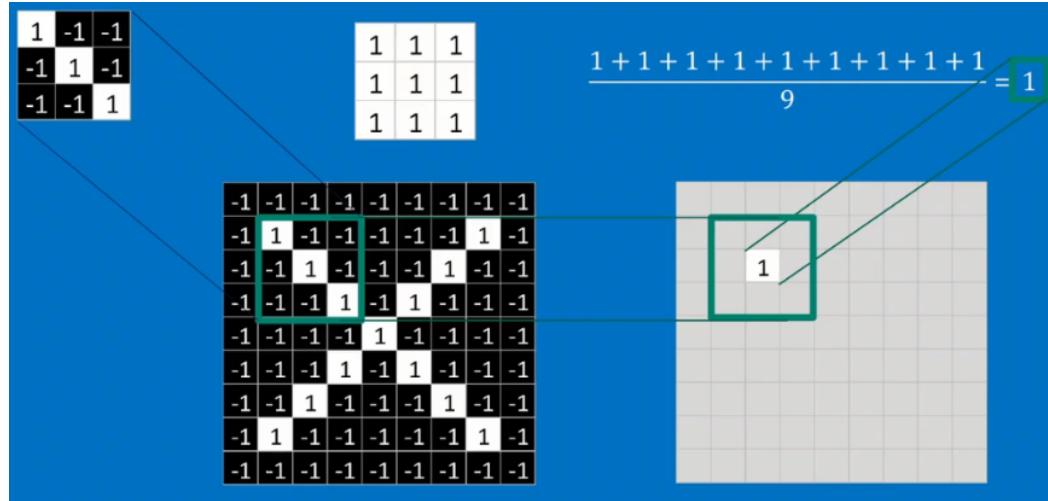
2.1 Kernel/Filter lị gợ? - Chic "ôn pin" soi mu

Trong th gii ca CNN, mòy tònñ hiu c mt bc nh, nú cn bit c nhng c trng njo lịm cho bc nh ú tr nññ c nht. Thay vø phón tòch tng im nh riõng l, mñnh s dng **Kernel** (hay Filter) - mt ma trn nh (thng lị

3×3 hoc 5×5). Kernel hot ng nh mt "ca s trt", di chuyn quõt qua tojn b bc nh tõm kim s tn ti ca cõc mu c th, vờ d nh mt ng cong, mt cnh dc, hay mt ng chõo. hiu rủ, mõnh i vio 1 vờ d luon nhõ.

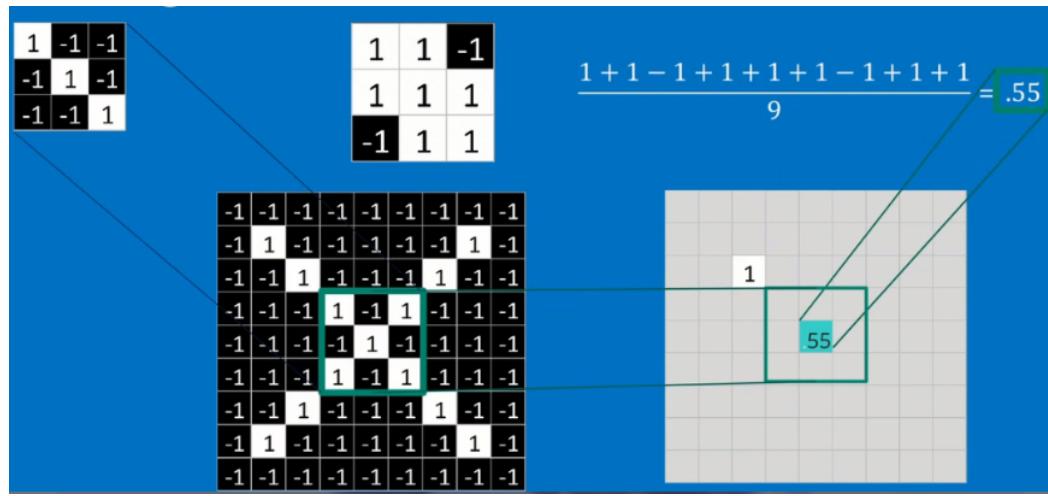
2.2 C ch khp mu (Feature Matching) - Bờ mt ca phõp tõch chp

Hõy hõnh dung mõnh mun dy mòy tõnh nhn din ch "X". Ch "X" c cu to bi hai ng chõo bt chõo nhau. Mt Kernel c thit k tõm "ng chõo trõi" s cú cõc giò tr dng (vờ d: 1) nm trõn ng chõo ú vị cõc giò tr óm (vờ d: -1) nhng v trõ cùn li. (Note: ma trn trng i din cho kt qu sau khi òp dng kernel)



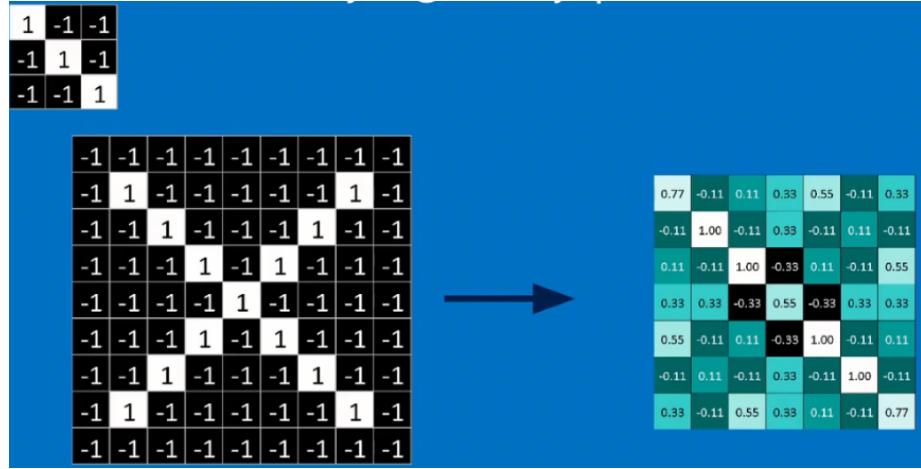
óy lị phn trng tóm hiu CNN t gc r. Thay vờ t duy theo kiu "kt ni tt c mi th" nh mng MLP truyñ thng, CNN tip cn hõnh nh theo cõch con ngi quan sòt: tõm kim cõc c trng (features).

Quò tronh "Filtering" (lc) din ra nh sau:



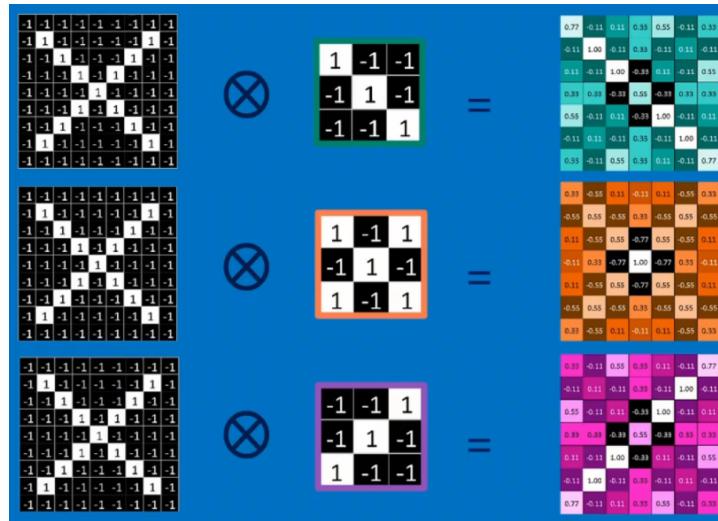
1. **Line up:** t Kernel chng lõn mt vúng nh cõng kõch thc.
2. **Multiply:** Nhón tng giò tr pixel ca nh vi giò tr tng ng trong Kernel (Element-wise multiplication).
3. **Add:** Cng tng tt c cõc kt qu li vị chia trung bõnh.

Kt qu:



- Nếu vùng nh bốn di thay đổi có hình ng chõo (kết hợp vi Kernel), phép nhân sẽ tạo ra cột số dung ln → Kt qu tng rt cao (vör d: 1.0 học 100%). Ta nói pattern ú õ c "kết hot" (activated).
- Nếu vùng nh khung khp (vör d: Kernel ng chõo nhng t lõn nhất vùng nh ng thng ng), còc phép nhân dung vị óm số tritt tiõu ln nhau → Kt qu tng xp x 0 học óm.
- Vör d tng t i vi nh 3D.**

Minh ha còc loi Filter khõc nhau trờch xut còc kiu c trng khõc nhau



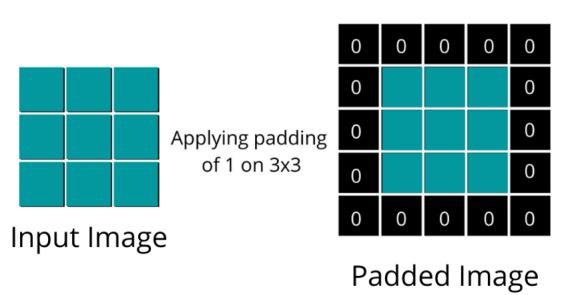
2.3 Còc phđp toòn kim soót B lc (Convolution Operation)

Stride (Bc nhý - ū vuñg kernel di chuyn bao nhieu bc mi lt): óy lị khong còch mị Kernel di chuyn sau mi ln tònh toòn. Nu $Stride = 1$, Kernel nhờch tng pixel mt, gi chi tit cao nht. Nhng nu ta tng $Stride > 1$ (vđ d: $Stride = 2$), Kernel s "nhý cúc" qua còc pixel. iu niy cù tòc dng gim kờch thc d liu u ra (Downsampling) ngay lp tc mị khñg cn lp Pooling, ging nh vic ta lt nhanh qua mt vn bn nm y chørnh thay vđ c tng ch.

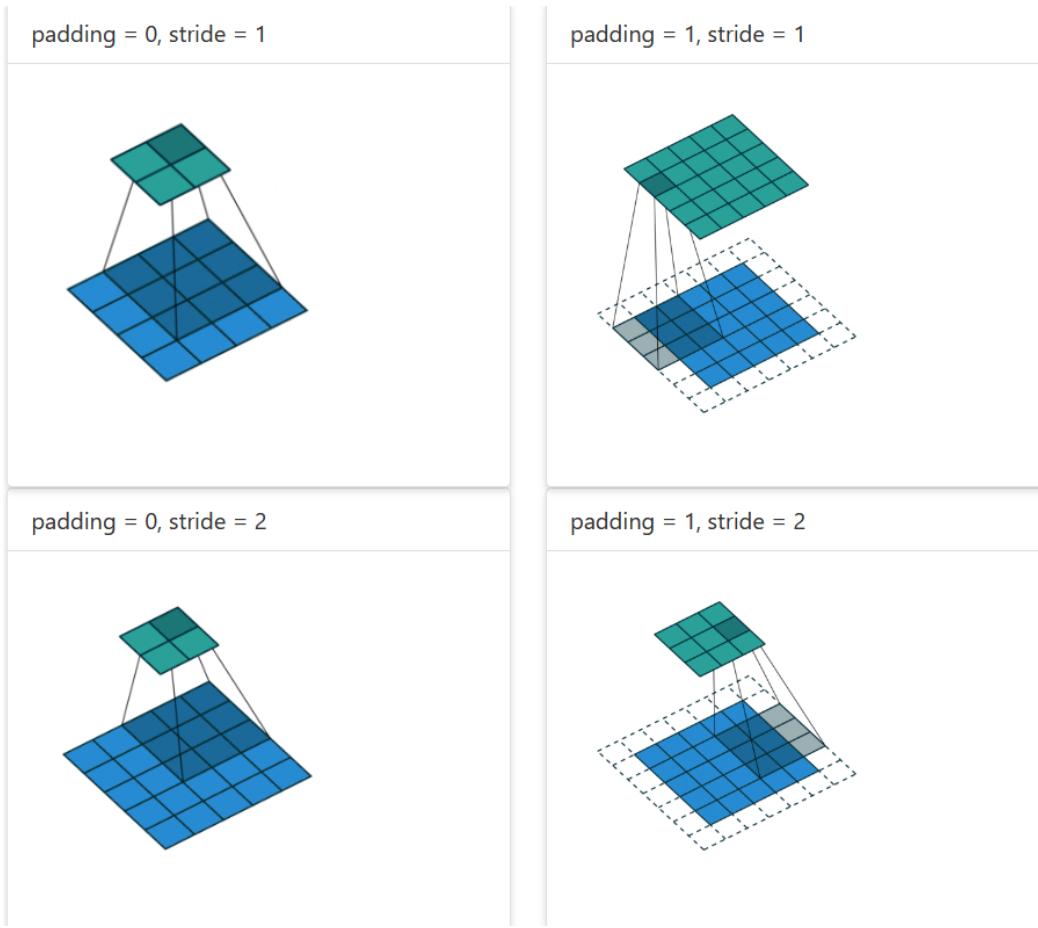
0	0	0	0	0	0	0
0	1	1	1	0	0	0
0	0	1	1	1	0	0
0	0	0	1	1	1	0
0	0	0	1	1	0	0
0	0	1	1	0	0	0
0	0	0	0	0	0	0

Figure 2: Vi phn in m lị tóm ca kernel, vi $stride = 2$, kernel s i 2 bc tònh t tóm. Nu kernel lị 2×2 tho ly pixel trồn cúng bốn trồi lim tóm

Padding (L - Pixel c bc thđm bđn ngoii nh): Khi Kernel trt røa bc nh, nū thng b thiу ht d liu (khñg kờch thc 3×3 nhón). iu niy dn n hai vn : nh b thu nh li sau mi lp tòch chp vi thñg tin røa nh b mt mót. Gii phòp lị **Zero-Padding**: thđm mt vin còc s 0 bao quanh nh gc. Lp "m" niy cho phđp Kernel t tóm ngay ti pixel ngoii cúng, giúp gi nguyễn kờch thc khñg gian ($Height \times Width$) ca Feature Map u ra.



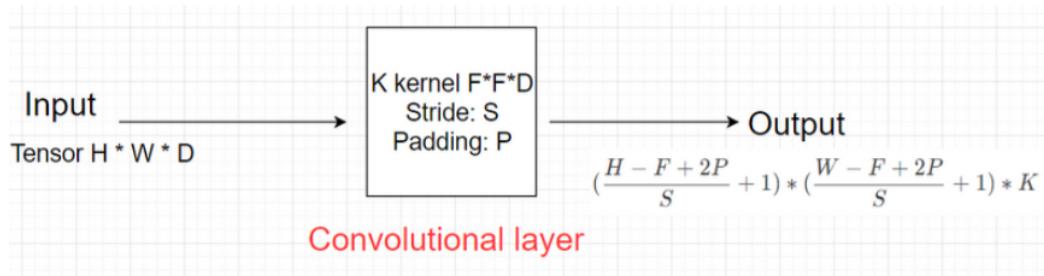
Minh ha Padding vđ Stride: Note: kờch thc nh hoc feature map (i.e. nh sau khi òp dng convolution) phi ln tng ng òp dng stride.



Cũng thc kôch thc u ra: thit k kin truc mng chòn xòc, ta cn tòn c kôch thc ca ma trn sau khi qua lp Conv:

$$Output = \frac{Input - Filter + 2 \times Padding}{Stride} + 1$$

Trong ú: *Input* lị kôch thc nh u vio, *Filter* lị kôch thc Kernel, *Padding* lị s lp vin thốm vio, vị *Stride* lị bc nhý.

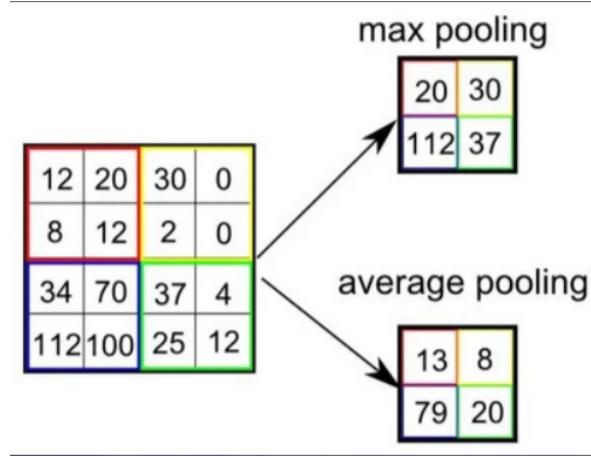


Vic trt Kernel qua nh khung phi lữc njo cng tu ý, nú c kim soot bi còc tham s tòn hc nghiôm ngt nh hñh d liu u ra.

Pooling (Gp c trng):

- **Max Pooling:** Ch gi li giò tr ln nht trong vúng (c trng ni bt nht), giüp nñn nh vị gim ti tòn nhng vn gi c thñg tin ct lñi.

- **Average Pooling:** Ly giò tr trung bnh ca tojn b còc pixel trong ving, nhm tng hp thñg tin mt cch mm, phn ònh mc hin din trung bnh ca c trng thay vch tp trung vio im mnht.



Phn 3: Kin truc CNN - T "Phón rô c trng" n "Hi ng b phiu"

Nu nh Kernel lị cung c tóm kim, tho Kin truc CNN chòn lị chin lc s dng nhng tóm kim ú. Bn cht ca CNN khñg phi lị nhơn thy ngay con móo hay chic xe hi, mì lị mt quò trønh **phón rô vị tòi to**.

3.1 Bn cht ca CNN: Chia tr

Thay vch c gng nhn din tojn b vt th ngay lp tc, nù chia nh biжи toòn thñh còc tng c trng (Feature Hierarchy) t n gin n phc tp:

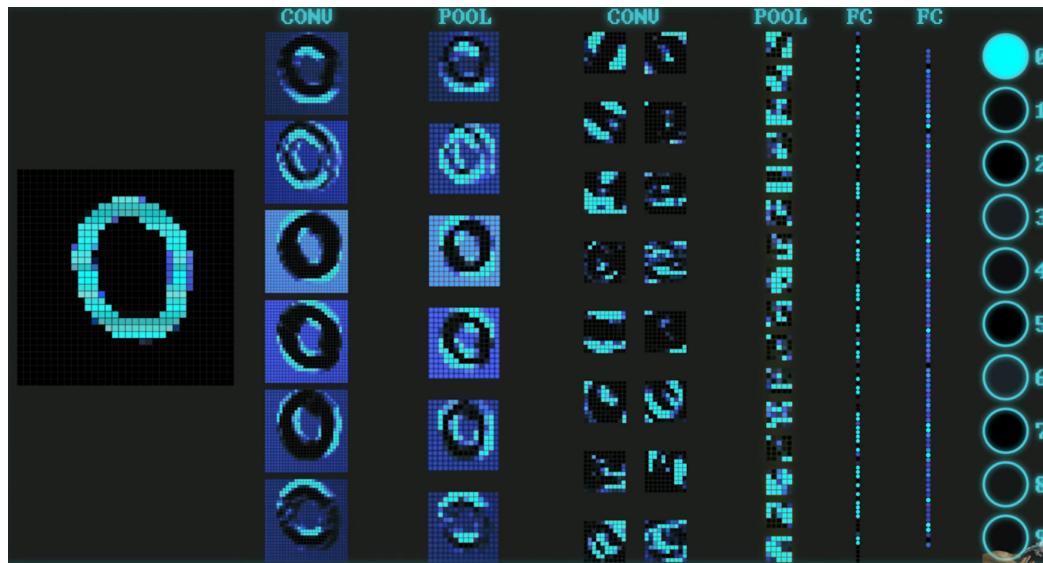


Figure 3: Minh ha s phón cp c trng trong CNN: T còc cnh n gin (tròi) n còc chi tit vt th phc tp (phi).

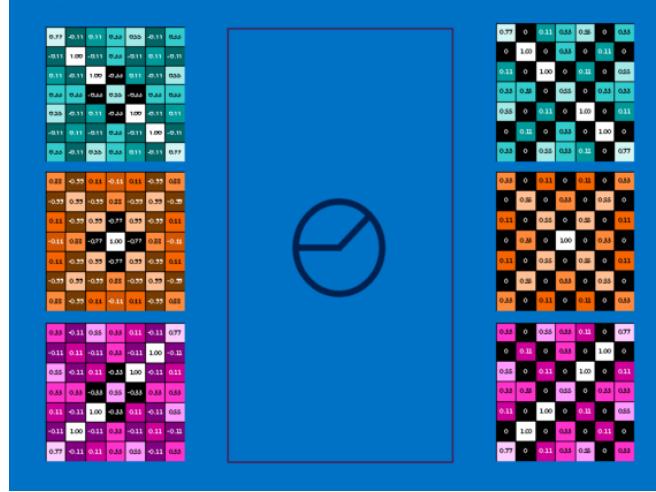
- **Tng thp (Low-level features):** còc lp u tiён, mng ch nhơn thy còc chi tit s khai nht nh **cnh, ng biển, mìu sc**. Trồn nh minh ha, chñng thng trñng ging nh còc vch k sc hoc nhiu.
- **Tng gia (Mid-level features):** Còc lp tip theo ghõp ni còc cnh nñy li thñh còc **ha tit, hñnh thứ c bn** nh ng cong, gúc vuñng, hñnh trùn (vò d: mt, bñnh xe, hoc mt phn ca con s).

- **Tng cao (High-level features):** Lp cui cung tng hp chung thinh ccc b phn hojn chnh nhn din vt th.

Mi lp úng vai trù nh mt b lc tinh vi hn lp trc ú, bin i ccc pixel vñ ngha thinh ccc c trng cù y ngha phón loi.

3.2 ReLU (Rectified Linear Unit) - Ngi gòc cng ca ccc lò phiu

Sau mi ln Kernel quõt qua nh (phõp tõch chp), chung ta thu c mt bn ccc giò tr. Cú ni giò tr rt dng (khp mu), cú ni giò tr óm (ngc mu). Læc niy, **ReLU** xut hin vi vai trù cc k quan trng nhng thng b hiu nhm lị ch "lịm toòn".



Theo gúc nhon bn cht, ReLU giúp mū hñnh tr li cùu hi: "**c trng niy cù xut hin hay khñng?**".

- Nu kt qu lị s dng: "Cú, tñi tñm thy mt cùi cnh óy!". Chung ta gi nguyễn giò tr ú.
- Nu kt qu lị s óm: "Khñng, hoc nñ ngc li vi cùi tñi tñm". ReLU s bin nñ thinh s 0.

Ti sao phi loi b s óm? Trong c ch b phiu (voting), chung ta ch quan tóm n s hin din ca ccc bng chng (evidence). Vic gi li ccc giò tr óm (nhiu/khñng khp) s lịm loõng thñng tin khi mnng c gng tng hp ccc c trng li vi nhau. ReLU giúp thanh lc tñm hiu, m bo rng ccc lp sau ch nhn c nhng "bng chng" chc chn nhnt.

3.3 Receptive Field - Tm nhon m rng

Khi chung ta xp chng (stack) ccc lp Convolution vñ Pooling lõn nhau, mt hin tng thñ v xy ra: **Receptive Field** (Vñng cm nhn) ca n-ron ngi y cing ln.

The diagram illustrates three separate matrix multiplication operations, each involving a large input matrix and a smaller mask matrix. The input matrix is a 7x7 grid of alternating 1s and -1s. The mask matrices are:

- Top row: $\begin{bmatrix} 1 & -1 & -1 \\ -1 & 1 & -1 \\ -1 & -1 & 1 \end{bmatrix}$
- Middle row: $\begin{bmatrix} 1 & -1 & 1 \\ -1 & 1 & -1 \\ -1 & -1 & 1 \end{bmatrix}$
- Bottom row: $\begin{bmatrix} -1 & 1 & 1 \\ 1 & -1 & 1 \\ 1 & 1 & -1 \end{bmatrix}$

The results of these multiplications are:

- Top result: A 7x7 matrix with values ranging from -0.77 to 0.77.
- Middle result: A 7x7 matrix with values ranging from -0.55 to 0.55.
- Bottom result: A 7x7 matrix with values ranging from -0.33 to 0.33.

This diagram shows a 7x7 input matrix with numerical values. The matrix is processed by a max pooling operation, which takes the maximum value from a 3x3 receptive field centered at each output unit. The resulting 3x3 output matrix is shown on the right, with values ranging from 0.11 to 1.00.

0.77	-0.11	0.11	0.33	0.55	-0.11	0.33
-0.11	1.00	-0.11	0.33	-0.11	0.11	-0.11
0.11	-0.11	1.00	-0.33	0.11	-0.11	0.55
0.33	0.33	-0.33	0.55	-0.33	0.33	0.33
0.55	-0.11	0.11	-0.33	1.00	-0.11	0.11
-0.11	0.11	-0.11	0.33	-0.11	1.00	-0.11
0.33	-0.11	0.55	0.33	0.11	-0.11	0.77

1.00	0.33	0.55	0.33
0.33	1.00	0.33	0.55
0.55	0.33	1.00	0.11

Rectified Linear Units (ReLUs)

This diagram shows the application of a Rectified Linear Unit (ReLU) activation function. The input matrix (left) contains both positive and negative values. The ReLU function sets all negative values to zero, resulting in the output matrix (right), where only the non-negative values from the input remain.

0.77	-0.11	0.11	0.33	0.55	-0.11	0.33
-0.11	1.00	-0.11	0.33	-0.11	0.11	-0.11
0.11	-0.11	1.00	-0.33	0.11	-0.11	0.55
0.33	0.33	-0.33	0.55	-0.33	0.33	0.33
0.55	-0.11	0.11	-0.33	1.00	-0.11	0.11
-0.11	0.11	-0.11	0.33	-0.11	1.00	-0.11
0.33	-0.11	0.55	0.33	0.11	-0.11	0.77

0.77	0	0.11	0.33	0.55	0	0.33
0	1.00	0	0.33	0	0.11	0
0.11	0	1.00	0	0.11	0	0.55
0.33	0.33	0	0.55	0	0.33	0.33
0.55	0	0.11	0	1.00	0	0.11
0	0.11	0	0.33	0	1.00	0
0.33	0	0.55	0.33	0.11	0	0.77

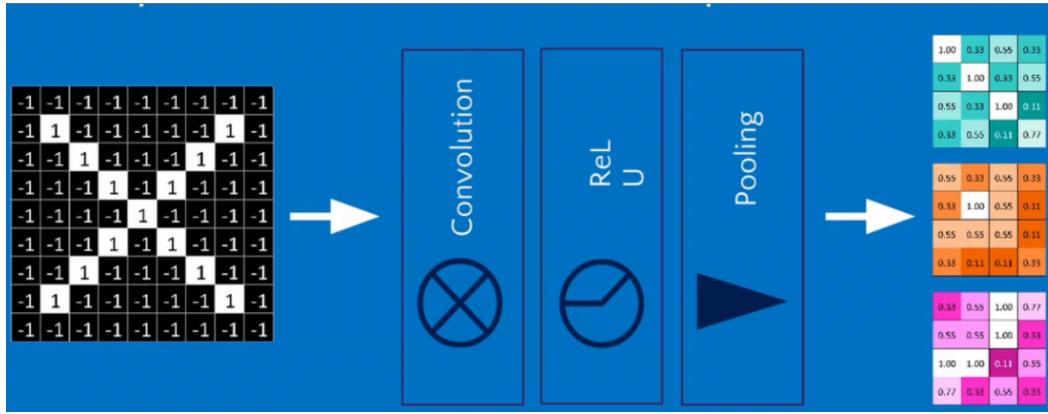
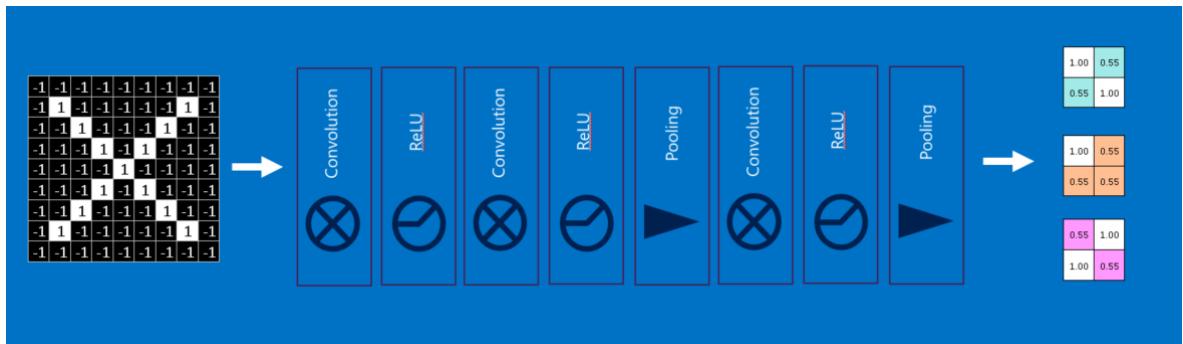


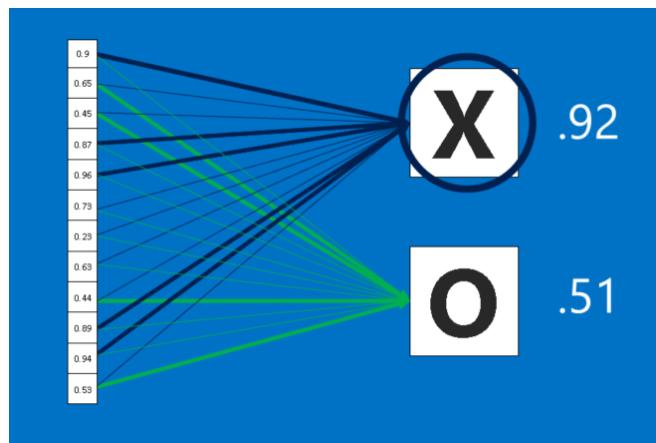
Figure 4: Mt im pixel lp Conv 1 ch i din cho vting 3×3 ca nh gc.



Nhng mt im pixel lp Conv 3 (sau khi qua Pooling) li i din cho vting 12×12 hoc ln hn ca nh gc. Chờnh nh c ch niy, cùc lp sóu (Deep Layers) cù th "nhơn" thy tng th bc nh a ra quyết nh, dứ chũng c xóy dng t nhng b lc rt nh ban u.

3.4 Gúc nhơn "Hi ng b phiu" (A List of Votes)

Cui cứng, sau khi i qua toin b quy tránh: Tờch chp (tóm c trng) \rightarrow ReLU (lc bng chng) \rightarrow Pooling (nỗn thũng tin), d liu t dng hñnh nh 3D s c đót phng (Flatten) thnh mt danh sòch đji cùc giò tr c trng cao cp. Lữc niy, lp Fully Connected (FC) úng vai trù nh mt "Hi ng b phiu", ni mi c trng tr thnh mt "c tri" a ra quyết nh cui cứng. Vờ d: "*Tui thy bñnh xe (giò tr cao) + Tui thy ca kñnh (giò tr cao)* \rightarrow *Tui b phiu cho lp: Ū TÙ*".



Mt cóu hi quan trng thng b b qua: Ti sao còc lp n niy, ta li u tiён dցng ReLU thay vờ him Sigmoid truyн thng? Cóu tr li nm kh nng "hc ngc" ca mng. Him Sigmoid nőn mi giò tr vio khong (0, 1), iu niy vў tinh khin tօn hiu o him b nh dn qua tng lp, dn n vn **Bin mt o him (Gradient Vanishing)**, ljm mng "quồn" mt còc c trng u vio. Ngc li, ReLU hot ng nh mt "cng dn truyн thng": vi còc giò tr dng, o him bng 1. iu niy khung ch khc phc trit vn bin mt o him mi cùn giўp tօn hiu **truy ngc (flip backward)** mt còc nguyễn vn t lp cui cúng v còc lp u. Nh ú, MLP cù th hc vи iu chnh còc mu (patterns) theo chiu nge li mt còc hiu qu, m bo rng mi liён kt gia "nguyễn nhón" (c trng nh) vи "kt qu" (nhón phón loi) luñ c duy trø cht ch trong sut quò trøn hun luyn.

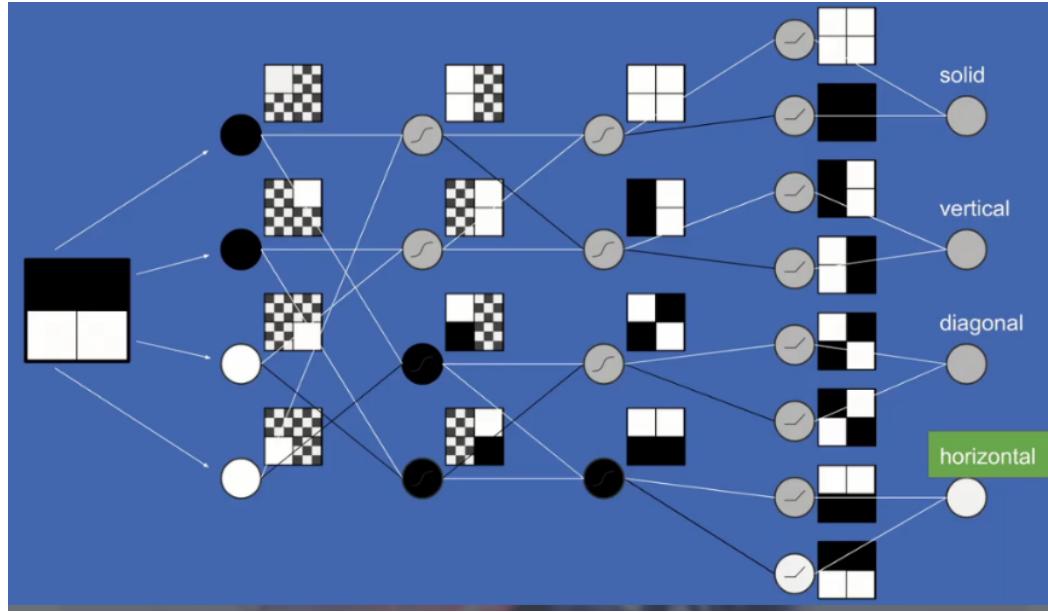


Figure 5: C ch b phiu ca lp Fully Connected: Còc c trng c tng hp a ra xòc sut phón loi.

Phn 4: CNN Backpropagation & S tht v phôp xoay Kernel

óy lị phn "khú nhn" nht nhng cng lị ni cha ng v p toòn hc ca CNN. Nu nh Forward Pass lị quò trøn møy tօn "nhn" bc nh, tho Backward Pass lị lữc nú "t kim im" rǔt kinh nghim.

Chñg ta s i sóu vio vic ljm th njo sai s (Loss) c truyн nge t lp cui cúng v còc lp u cp nht b lc (Kernel).

4.1 Mc tiўu ca chñg ta

Gi s ta cù mt lp Convolution n gin:

$$Y = X * K$$

Trong ú:

- X : nh u vio (Input).
- K : B lc (Kernel/Filter).
- Y : Bn c trng u ra (Output Feature Map).
- L : Hjm mt mót (Loss Function) cui cúng ca mng.

Nhim v ca Backpropagation lị tօm ra hai o him riўng (Gradients) quan trng:

1. $\frac{\partial L}{\partial K}$: cp nht trng s ca b lc (giúp mòi hc c c trng tt hn).
2. $\frac{\partial L}{\partial X}$: truyn sai s v cho lp ng trc nú (tip tc chui Backprop).

4.2 Tờnh o hịm cho B lc ($\frac{\partial L}{\partial K}$)

Tng tng Kernel K ang trt trőn nh X . Mi ln trt, nú to ra mt giò tr trőn Y . Vợ vy, mt trng s trong Kernel s úng gúp viø tt c còc pixel u ra mi nú i qua.

Theo quy tc chui (Chain Rule), o hịm ca li i vi mt trng s w trong Kernel lị tng hp ca tt c còc li mi trng s ú góy ra trőn toin b bn u ra Y :

$$\frac{\partial L}{\partial K_{ij}} = \sum_m \sum_n \frac{\partial L}{\partial Y_{mn}} \cdot \frac{\partial Y_{mn}}{\partial K_{ij}}$$

iu thř v lị: Khi bn trin khai cung thc niy ra, nú chørnh lị mt phđp Cross-Correlation gia nh u viø X vị bn li $\frac{\partial L}{\partial Y}$.

Trc giòc: bit nöñ sa Kernel th njo, ta ly "Bn li" m lön "nh gc". Ni njo nh gc cú giò tr ln mi li cng ln, ngha lị Kernel ti ú cn c iu chnh mnh tay nht.

4.3 Tờnh o hịm cho u viø ($\frac{\partial L}{\partial X}$) - S xut hin ca Rot180

óy lị phn góy lř ln nht vị cng lị lỳ do ti sao nhip tři liu toòn hc li nhc n vic "xoay 180".

tờnh gradient cho lp u viø X (nhm truyn v lp trc), ta cn bit mi pixel x_{ij} õ nh hng n bao nhiu pixel trőn u ra Y . Theo toòn hc, cung thc tờnh o hịm niy chørnh lị phđp tờch chp (Convolution) gia bn li $\frac{\partial L}{\partial Y}$ (c lút thřm vin - padded) vi Kernel õ b xoay 180 ($rot180(K)$).

$$\frac{\partial L}{\partial X} = \frac{\partial L}{\partial Y} * rot180(K)$$

Ti sao phi xoay?

- Trong quò tronh Forward (i ti), pixel x_{11} nhón vi k_{11} .
- Nhng khi Backward (i lři), truy ngc dùng chy t Y v X , còc mi liён h khřng gian b o ngc. Phn t gúc trőn-tròi ca Kernel lřc i ti thc cht li tng ng vi phn t gúc di-phi khi ta xôt t gúc lan truyn ngc.

4.4 S tht v Cross-Correlation vs. Convolution

Trong còc th vin Deep Learning thc t (nh PyTorch hay TensorFlow), cú mt s khòc bit nh gia lỳ thuyt vị thc hinh:

- Lỳ thuyt Toòn hc: Phđp Convolution chun ($f * g$) yđu cu phi lt ngc Kernel 180 trc khi trt.
- Thc t Code: Hu ht còc th vin s dng phđp Cross-Correlation (trt mi KHŘNG lt) cho quò tronh Forward Pass gim chi phò tờnh toòn.

Chørnh vđ Forward Pass dřng Cross-Correlation (khřng xoay), nöñ m bo tờnh ũng n ca o hịm, quò tronh Backward Pass bt buc phi thc hin phđp toòn ngc li - chørnh lị Convolution thc s (tc lị Cross-Correlation vi Kernel xoay 180).

Tóm li quy trnh chun:

- **Forward:** Trt Kernel thng (Cross-Correlation).
- **Backward (tờnh cho X):** Trt Kernel xoay 180 (Convolution chun).

óy lị v p ca s i xng trong toòn hc: Anh i ti bng còch trt xuūi, thø anh phi i lúi bng còch trt ngc mi th khp li hojn ho.

Phn 5: CNN Nóng cao - 1x1 Convolution & Phón bit chiu khñg gian

Mrng kin the sang còc khòi nim hin i vị sa cha còc hiu lm ph bin.

- **Phón bit 2D vs 3D CNN:**

- Lịm rủ hiu lm: nh mịu RGB (3 kñnh) vn dúng 2D CNN (x lỳ khñg gian Height-Width).
- 3D CNN thc s dúng cho d liu video (thòm chiu thi gian/Temporal).

- **Phõp mịu ca 1×1 Convolution:**

- Nú hot ng nh mt "Single Neuron" òp dng lzn chiu sóu (channels) ca tng pixel.
- ng dng: Gim s chiu d liu (Dimensionality Reduction) vị trn thñng tin gia còc kñnh mịu (Channel Pooling) mị khñng thay i kòch thc khñng gian.