

Formal Proof of 2D Convolution and Spatial Sum-Pooling Fusion

HEIR Compiler Team

June 25, 2026

1 Introduction

This document provides a formal mathematical proof of the equivalence between a cascade of a 2D convolution followed by a 2D spatial sum-pooling layer, and a single, equivalent 2D convolution layer.

Nb., this document was generated by Gemini, and reviewed by Jeremy Kun.

2 Definitions

Let:

- $I \in \mathbb{R}^{C \times H_{in} \times W_{in}}$ be the input tensor (batch size $N = 1$ is assumed and omitted for clarity).
- $W \in \mathbb{R}^{F \times C \times K_H \times K_W}$ be the convolution filter weights, where F is the number of output channels, C is the number of input channels, and $K_H \times K_W$ is the kernel shape.
- $Y \in \mathbb{R}^{F \times H_{conv} \times W_{conv}}$ be the output of the convolution layer.
- $Z \in \mathbb{R}^{F \times H_{out} \times W_{out}}$ be the output of the pooling layer.
- $s_c = (s_{c,H}, s_{c,W})$ be the convolution strides.
- $d_c = (d_{c,H}, d_{c,W})$ be the convolution dilations.
- $P = (P_H, P_W)$ be the pooling window shape.
- $s_p = (s_{p,H}, s_{p,W})$ be the pooling strides.
- $d_p = (d_{p,H}, d_{p,W})$ be the pooling dilations.

We assume pooling padding is zero ($p_{pool} = 0$).

3 Mathematical Formulation

3.1 Convolution Layer

The output of the 2D convolution for channel f at spatial coordinates (h_y, w_y) is defined as:

$$Y(f, h_y, w_y) = \sum_{c=0}^{C-1} \sum_{kh=0}^{K_H-1} \sum_{kw=0}^{K_W-1} I(c, h_y \cdot s_{c,H} + kh \cdot d_{c,H}, w_y \cdot s_{c,W} + kw \cdot d_{c,W}) \cdot W(f, c, kh, kw) \quad (1)$$

for $h_y \in [0, H_{conv} - 1]$ and $w_y \in [0, W_{conv} - 1]$.

3.2 Sum-Pooling Layer

The output of the sum-pooling layer for channel f at spatial coordinates (h_z, w_z) is defined as:

$$Z(f, h_z, w_z) = \sum_{ph=0}^{P_H-1} \sum_{pw=0}^{P_W-1} Y(f, h_z \cdot s_{p,H} + ph \cdot d_{p,H}, w_z \cdot s_{p,W} + pw \cdot d_{p,W}) \quad (2)$$

for $h_z \in [0, H_{out} - 1]$ and $w_z \in [0, W_{out} - 1]$.

4 Derivation of Equivalence

We substitute the convolution equation (1) into the pooling equation (2):

$$Z(f, h_z, w_z) = \sum_{ph=0}^{P_H-1} \sum_{pw=0}^{P_W-1} \left[\sum_{c=0}^{C-1} \sum_{kh=0}^{K_H-1} \sum_{kw=0}^{K_W-1} I(c, (h_z \cdot s_{p,H} + ph \cdot d_{p,H}) \cdot s_{c,H} + kh \cdot d_{c,H}, (w_z \cdot s_{p,W} + pw \cdot d_{p,W}) \cdot s_{c,W} + kw \cdot d_{c,W}) \cdot W(f, c, kh, kw) \right] \quad (3)$$

By linearity of summation, we can swap the order of summation, bringing the input channel c and the spatial loops to the outside:

$$Z(f, h_z, w_z) = \sum_{c=0}^{C-1} \sum_{ph=0}^{P_H-1} \sum_{pw=0}^{P_W-1} \sum_{kh=0}^{K_H-1} \sum_{kw=0}^{K_W-1} I(c, h_z \cdot s_{p,H} \cdot s_{c,H} + ph \cdot d_{p,H} \cdot s_{c,H} + kh \cdot d_{c,H}, w_z \cdot s_{p,W} \cdot s_{c,W} + pw \cdot d_{p,W} \cdot s_{c,W} + kw \cdot d_{c,W}) \cdot W(f, c, kh, kw) \quad (4)$$

We define the **effective stride** $s_{eff} = (s_{eff,H}, s_{eff,W})$ as:

$$s_{eff,H} = s_{p,H} \cdot s_{c,H}, \quad s_{eff,W} = s_{p,W} \cdot s_{c,W} \quad (5)$$

Let the effective spatial offsets in the input tensor be:

$$u(ph, kh) = ph \cdot d_{p,H} \cdot s_{c,H} + kh \cdot d_{c,H} \quad (6)$$

$$v(pw, kw) = pw \cdot d_{p,W} \cdot s_{c,W} + kw \cdot d_{c,W} \quad (7)$$

Now we can rewrite the input index in (4) using s_{eff} and (u, v) :

$$Z(f, h_z, w_z) = \sum_{c=0}^{C-1} \sum_{ph=0}^{P_H-1} \sum_{pw=0}^{P_W-1} \sum_{kh=0}^{K_H-1} \sum_{kw=0}^{K_W-1} \left[I(c, h_z \cdot s_{eff,H} + u(ph, kh), w_z \cdot s_{eff,W} + v(pw, kw)) \cdot W(f, c, kh, kw) \right]$$

We want to express this in the form of a single convolution with effective filter W_{eff} :

$$Z(f, h_z, w_z) = \sum_{c=0}^{C-1} \sum_{u=0}^{K_{eff,H}-1} \sum_{v=0}^{K_{eff,W}-1} I(c, h_z \cdot s_{eff,H} + u, w_z \cdot s_{eff,W} + v) \cdot W_{eff}(f, c, u, v) \quad (8)$$

To determine the bounds of u and v (the effective kernel size K_{eff}), we find the maximum values of $u(ph, kh)$ and $v(pw, kw)$:

$$K_{eff,H} = \max(u(ph, kh)) + 1 = (P_H - 1) \cdot d_{p,H} \cdot s_{c,H} + (K_H - 1) \cdot d_{c,H} + 1 \quad (9)$$

$$K_{eff,W} = \max(v(pw, kw)) + 1 = (P_W - 1) \cdot d_{p,W} \cdot s_{c,W} + (K_W - 1) \cdot d_{c,W} + 1 \quad (10)$$

We can define W_{eff} using Iverson brackets to map the coordinates:

$$W_{eff}(f, c, u, v) = \sum_{ph=0}^{P_H-1} \sum_{pw=0}^{P_W-1} \sum_{kh=0}^{K_H-1} \sum_{kw=0}^{K_W-1} W(f, c, kh, kw) \cdot [u = u(ph, kh)] \cdot [v = v(pw, kw)] \quad (11)$$

This is precisely the weight convolution loop implemented in the compiler:

$$W_{eff}(f, c, u(ph, kh), v(pw, kw)) \leftarrow W_{eff}(f, c, u(ph, kh), v(pw, kw)) + W(f, c, kh, kw) \quad (12)$$

initialized to zero. This proves the equivalence to a single convolution.