

Attendance Project

G H Lathrom

May 16, 2021

1 Overview

This project explores the possibility of using image recognition with photos from student ids to perform automatic attendance checks. The plan is to use OpenCV to process images and to write the code in C++. This is mainly because I want to write in a language other than Python in addition to speed being a factor.

2 Basics of OpenCV

The mathematical operation of **convolution** is a mathematical operation which takes in two functions, f and g and returns another function $f * g$ defined as

$$(f * g)(t) = \int_{-\infty}^{\infty} f(\tau)g(t - \tau) d\tau$$

and satisfies the following properties:

- (i) Commutativity: $f * g = g * f$
- (ii) Associativity: $f * (g * h) = (f * g) * h$
- (iii) Distributivity: $f * (g + h) = f * g + f * h$
- (iv) Associativity of scalars: $\alpha(f * g) = (\alpha f) * g$.

Using the **distribution** or **generalized function**, the Dirac function, δ , we can define an identity element

$$f * \delta = f$$

With respect to differentiation and integration we have

$$(f * g)' = f' * g = f * g' \quad \text{and} \quad \int (f * g) dx = \left(\int f dx \right) \left(\int g dx \right).$$

The discrete convolution of two functions is defined as

$$(f * g)(n) = \sum_{m=-\infty}^{\infty} f(m)g(n-m).$$

In the case of an image consisting of n -rows and m -columns, we may take the image as a function

$$f : \{0, \dots, n-1\} \times \{0, \dots, m-1\} \rightarrow \mathbb{R}$$

in the case a grey-scale image or

$$f : \{0, \dots, n-1\} \times \{0, \dots, m-1\} \rightarrow \mathbb{R}^3$$

in the case of a color image with red-green-blue pixel triples. We can extend these functions to functions of on all of \mathbb{Z} in multiple ways, but here I mention a couple of common ways. The first is to extend the function f periodically to a function \hat{f} , that is we say

$$\hat{f}(i, j) = f(i \bmod n, j \bmod m)$$

where we have taken the standard representation sets $\{0, \dots, n-1\}$ and $\{0, \dots, m-1\}$. This would take an image to periodic image as shown in Figure 1. Another extension is to simply make the values zero outside of the original image values as

$$\hat{f}(i, j) = \begin{cases} f(i, j), & (i, j) \in \{0, \dots, n-1\} \times \{0, \dots, m-1\} \\ 0 & \text{otherwise} \end{cases}$$

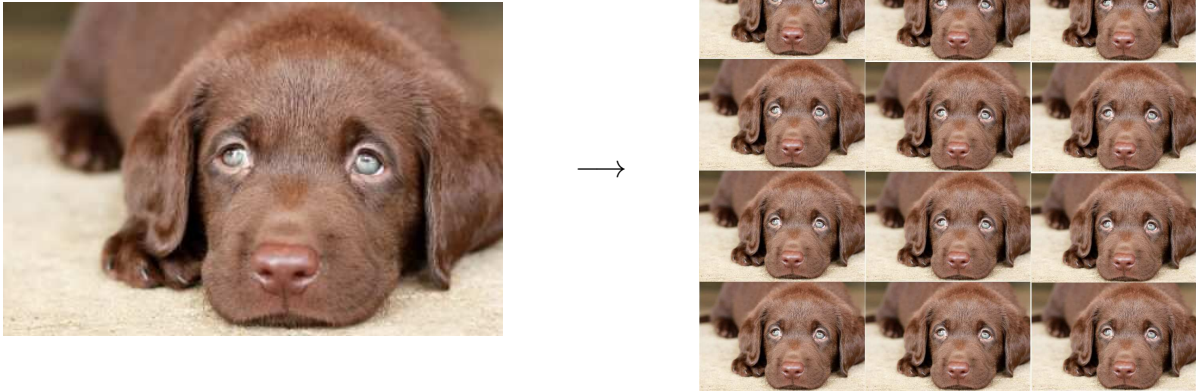


Figure 1: Periodic Boundary

[Rud76]

References

[Rud76] W. Rudin, *Principles of mathematical analysis*, 3rd ed., McGraw-Hill, 1976.