**Título de tesis:** Diseño de un procesador para resolver la operación de la convolución mediante el método de Systolic Arrays.

**Capítulo 1 – Anteproyecto**

1. **Antecedentes**

**Hablar de cómo todos los sistemas de visión necesitan realizar la convolución para procesar la imagen.**

Los sistemas embebidos son sistemas de computación que normalmente se encarga de realizar una tarea en específico en donde tienen *time constrains* de tiempo real. En esto difieren de los sistemas computacionales de uso general, que ambos tiene que realizar una tarea y producir resultados correctos, pero el sistema embebido además tiene que realizarlo en tiempo para cumplir con *deadlines*. Una de las características deseada de un sistema embebido es un bajo consumo de potencia. La forma en que esta característica se logra es mediante la utilización de microcontroladores menos potentes/rápidos comparados con lo que corren otrosn sistemas como los servidores. Una técnica muy utilizada para

En la actualidad se utilizan los SoC (Systems on Chip), el cual es un circuito integrado que integra los componentes de un

Un coprocesador es…

La convolucion se define en forma matematica como la combinacion de dos se;ales para producir una tercer se;al… APLICACIONES de la convolucion (procesamiento de imagenes).

Es la tecnica mas importante en el procesamiento digital de se;ales, donde los sistemas se describen mediante una se;al llamada respuesta al impulso. La convolucion es importante debido a que relaciona las tres se;ales de interes: la se;al de entrada, la se;al de salida y la respuesta al impulso, y en un sistema lineal si se conoce su respuesta al impulso, se conoce el comporatmiento del mismo ante cualquier se;al de entrada, lo que nos ayuda a controlar el sistema.

ARREGLAR – Este tipo de computo es intensivo, es decir, requiere que se realicen demasiadas operaciones matematicas. En la actualidad, se utiliza la computacion acelerada por medio del uso de GPU, el cual consiste en hacer *processing offloading* al CPU y pasar carga de procesamiento a los GPU.

Las tecnologías infraestructurales son aquellas que cambian la perspectiva de la innovación e investigación, siendo una de ellas la Inteligencia Artificial. En la actualidad se está dando una revolución en el enfoque de la computación debido a la creciente cantidad de datos con la que se cuenta y a las técnicas que se utilizan para analizar los mismos. Datos son cantidades, caracteres o símbolos en los cuales se realizan operaciones computacionales, se almacenan y transmiten en forma de señales eléctricas y se graban en medios de grabación. Se calcula que para el año 2020 en el mundo habrá 50 mil millones de dispositivos conectados a la red, todos estos dispositivos generando datos [1].

Asimismo, existe una subcategoría de la Inteligencia Artificial llamada Redes Neuronales, que es un paradigma de programación inspirado en las neuronas del cerebro que le permite a una computadora aprender de datos observados, y el Aprendizaje Profundo que es un conjunto de técnicas para aprender en base a redes neuronales [2].

Existe una amplia gama de aplicaciones donde se emplean técnicas de Aprendizaje Profundo. En robótica donde se utilizan sistemas de visión por computadora para carros autónomos, en internet y la nube donde se utiliza para reconocimiento del habla, sistemas de recomendación y clasificación de imágenes, en medicina y biología donde hay sistemas de detección de células cancerígenas y en seguridad y defensa donde hay sistemas de detección facial y video vigilancia. Muchas de las aplicaciones consisten en el reconocimiento de imágenes.

Las red neuronal convolucional (CNN) es una técnica que forma parte del Aprendizaje Profundo y se utiliza ampliamente para el reconocimiento de imágenes. Consiste en una capa de entrada y salida, así como de múltiples capas internas. Estas capas ocultas suelen ser capas convolucionales, capas de agrupación, capas totalmente conectadas y capas de normalización. En la capa convolucional, se aplica una operación de convolución a la entrada, pasando el resultado a la siguiente capa [3].

La convolución es una forma matemática de combinar dos señales para formar una tercera señal. Es la técnica más importante en el procesamiento digital de señales, donde los sistemas se describen mediante una señal llamada respuesta al impulso. La convolución es importante porque relaciona las tres señales de interés: la señal de entrada, la señal de salida y la respuesta al impulso [4].

    Este tipo de cómputo es intensivo, es decir requiere que se realicen demasiadas operaciones matemáticas. En la actualidad, se utiliza la computación acelerada por GPU, que es el uso de GPU (graphics processing unit) junto a un CPU (central processing unit) para acelerar las aplicaciones de Aprendizaje Profundo [5].

[Borghesi] Image-based techniques currently represent a significant field of research – computer vision, robot vision, remote sensing, acoustical imaging and topography are a few examples of fields in which data acquired from the external world are organized as images.

In these fields, there are two main classes of computations:

1) Image formation process – in which sparse data are processed to organize them in a conventional image form.

2) Image processing - The case where data acquired already represent an image of the external world and the computer target consists of extracting, describing, and understanding the information contained in the image.

In both cases a large number of image transformation techniques are based on convolution algorithms,1 which, in spite of their extremely useful mathematical properties, are too slow to permit real applications on general-purpose computers.

1. **Estado del arte**

Google TPU

La ley de Moore se esta muriendo.

1. **Planteamiento del Problema**

Von Neumann bottlenecks – checar paper Kung

Las aplicaciones de computo son cada vez mas intensas por lo tanto se necesita hardware mas eficiente.

La ley de Moore se esta muriendo.

Cada vez mas dispositivos son capaces de capturar imagenes.

Actualizar TODO ESTO - A medida que la capacidad de los sistemas electronicos que recopilan datos aumenta, la necesidad para procesar los mismos tambien aumenta. Los GPU son circuitos electronicos especializados que ayudan al computo intensivo, pero trabajan a nivel de software, lo cual no los hace la opcion mas rapida en cuestion de procesamiento de datos, ademas de consumir mucha potencia, lo cual resulta como una desventaja para sistemas que utilizn baterias. Existe la necesisdad de encontrar alternativas a las tendencias actuales para el procesmaiento de se;ales con mejores presatciones de computo intensivo yu con menor consumo de potencia.

A medida de que nuestra capacidad de recopilar datos aumenta, la obligación para analizar los mismos también aumenta. Los GPUs son circuitos electrónicos especializados que ayudan al cómputo intensivo, pero trabajan a nivel de software, lo cual los hace evidentemente no la opción más rápida en cuestión de procesamiento de datos, además de consumir mucha potencia. Actualmente existe la necesidad de encontrar alternativas a las tendencias actuales para el procesamiento de señales con mejores prestaciones de cómputo intensivo y con menor consumo de potencia.

No existe un IP block de Xilinx que haga la convolución

1. **Objetivos**

Desarrollar un coprocesador capaz de resolver la convolución, que sea compacto, de bajo consumo de potencia.

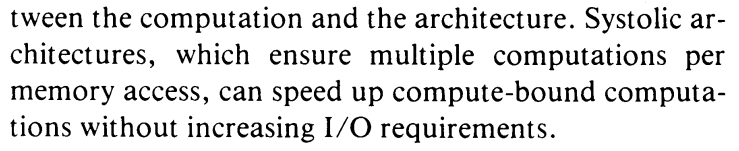
Desarrollar un coprocesador para la operaci[on de la convolucion discreta. Describir una arquitectura compacta y de bajo consumo de energia con el lenguaje de descripcion Verilog e implementar en un FPGA, que pueda utilizarse como acelerador de hardware en sistemas que hagan uso intensivo de dicha operación (como celulares o cualquier dispositivo que trabaje con imagenes)

Desarrollar un procesador en FPGA que sea dedicado que ejecute la convolución en base a técnicas de arquitectura digital actuales que sea eficiente a nivel de hardware y con un consumo bajo de potencia, que se pueda utilizar como acelerador de cómputo en sistemas que utilicen la función matemática de convolución.

1.4 Hipotesis

La arquitectura digital para la convolución implementada en el FPGA tendrá un un desempeño mejor o comparable mejor desempeño en t{erminos de para efectos de tiempos de procesamiento que un acelerador con GPU.

1. **Hipotesis**

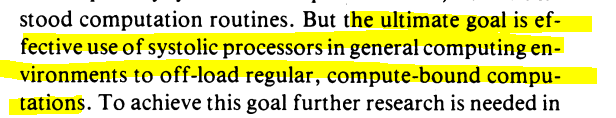


1. **Justificación**

[Borghesi] The computational regularity of convolution can be exploited for efficient design of high-throughput architecture able considerably to decrease the calcuolation times with respect to those typical of full-software solutions.

Se necesitan diseños de coprocesadores que ayuden a quitarle sobrecarga a los procesadores.

El motivo para elegir una systolic array es porque el diseño sirve para quitarle payload al CPU, pero además es una arquitectura que resulta de uso general. - Kung



Existen diversos dispositivos para acelerar el procesamiento de datos. El dispositivo mas popular para esta tarea es el Graphic Processing Unit GPU, el cual ofrece una alta velocidad de procesamiento de imagenes, pero tiene la desventaja de ser muy demandante de potencia, es por esto, que este tipo de arquitectura no sirve para dispositivos mobiles o sistemas embebidos (otra desventaja de los GPUS es que los lenguajes para programarlos en su mayoria son complejos y no hay muchos programadores que los dominen). Es por esto, que una arquitectura desarrollada en un FPGA (que eventualmente se puede transformar en un chip dedicado de hardware) brinda la oportunidad de tener tiempos de procesamiento similares (o mayores porque es hardware dedicado), adem[as con un menor consumo de potencia, lo cual lo hace adecuado para los sitemas mobiles.

A pesar de que en la actualidad existen opciones de dispositivos para acelerar el procesamiento de datos, como el GPU, este no está pensado para bajos consumos de potencia y al ser de propósito general, no está optimizado para la función convoluciónpara sistemas que requieran movilidad en específico. Por lo tanto, con una esta arquitectura en FPGA se logrará hacer un (co)procesador que esté dedicado para esta función disminuya lo tiempos de procesamiento en el caso de y que ayude en la industria cuando se necesite procesar grandes volumenes cantidades de datos. En caso de no realizarse esta arquitectura se estaría desaprovechando una oportunidad de analizar todos los datos con los que se cuentan hoy en día de manera más eficiente, además de que se estaría gastando mayor cantidad de potencia que representa se traduce en un gasto económico.

1. **Delimitaciones**

[Borghesi] We describe a fast coprocessor able to perform and- and two- dimensional convolutions.

Este proyecto se limitará a implementar la arquitectura para un coprocesador capaz de resolver la convolución. No se implementarán las interfaces para que pueda ser parte de un SoC o un sistema más grande.

Este trabajo se limitara al dise;o de un coprocesador para la operaci[on de la convolucion discreta. La arquitectura se desarrollara con el lenguaje de descripci[on de hardware Verilog y se verificara a traves de un entorno de simulacion en el lenguaje C++.

El diseño del procesador se limitará a implementar la arquitectura para un (co) procesador para la convolución. Se implementará en el FPGA y se probaráverficará a través de un entorno de simulación con la ayuda con un script de Matlab. para checar el rendimiento.

1. **Limitaciones**

Las principales limitaciones del proyecto son:

- Tiempo disponible

- Dinero

- Recursos – Equipo de hardware (FPGAs), licencias

- Licencias No se podrá comparar el rendimiento y el consumo de potencia de la arquitectura del procesador para la convolución implementada en el FPGA con el rendimiento en un GPU, esto debido a que no se cuenta con una GPU para implementar el algoritmo con el que se probará la arquitectura en FPGA.

**Bibliografía**

[1] Nordrum A. (2016). Popular Internet of Things Forecast of 50 Billion Devices by 2020 Is Outdated. 13/02/2018, de IEEE Spectrum Sitio web: https://spectrum.ieee.org/tech-talk/telecom/internet/popular-internet-of-things-forecast-of-50-billion-devices-by-2020-is-outdated

[2] Mills T. (2018). Deep Learning and Neural Networks. 13/02/2018, de Forbes Sitio web: https://www.forbes.com/sites/forbestechcouncil/2018/01/02/deep-learning-and-neural-networks/#3d6e5af939d8

[3] UFLDL Stanford. (2017). Convolutional Neural Network. 13/02/2018, de Stanford Sitio web:

http://ufldl.stanford.edu/tutorial/supervised/ConvolutionalNeuralNetwork/

[4] Smith S. W. (1999). The Scientist and Engineer's Guide to Digital Signal Processing. USA: California Technical Publishing.

[5] NVIDIA. (2017). What is GPU-Accelerated computing? 13/02/2018, de Artificial Intelligence Computing Leadership from NVIDIA. Sitio web: <http://www.nvidia.com/object/what-is-gpu-computing.html>

[Borghesi] *Paper - Programmable modified systolic array for fast one- and two-dimensional convolutions*

[DSP Book in Drive] - The Scientist and Engineer's Guide to Digital Signal Processing

**Proyecto**

Source - https://en.wikipedia.org/wiki/Kernel\_(image\_processing)#Convolution

Hacer un convolucionador para realizar procesamiento de imagen con hardware.

Yo sé que esto se puede hacer mediante software, pero si utilizamos hardware, esto lo va a hacer más rápido, más eficiente en energía (maybe?)

Mi idea es la siguiente, tener un sistema que realice la convolución (el hardware que desarrolle yo), el cual sea capaz de realizar diferentes transformaciones a una imagen de entrada.

Para esto, necesito un convolucionador 2D.

Input: Una imagen - lo ideal sería tener un 1080p (1920x1080), pero por cuestiones de recursos, probablemente sea demasiado grande, entonces maybe 720p (que todavía creo que es muy grande). La imagen debería de estar en escala de grises. Puedes iniciar con una imagen 8x8 para probar que tu algoritmo funciona.

Dentro del sistema, tendré filtros conocidos para realizar operaciones como *blurring, sharpening, embossing, edge detection, etc.* Para iniciar, podría empezar con el kernel más sencillo - me gusta el de edge detection.

Output: la imagen modificada. En escala de grises.

El beneficio de utilizar este tipo hardware sobre un sistema que solo utiliza un CPU, es que no necesitamos un CPU tan poderoso, además esto ayuda a preparar la imagen para su futuro análisis con técnicas más avanzadas de Computer Vision por ejemplo.

Mi circuito tiene que ser secuencial, ya que la convolución es una especie de suma. Mi circuito tiene que ser síncrono.

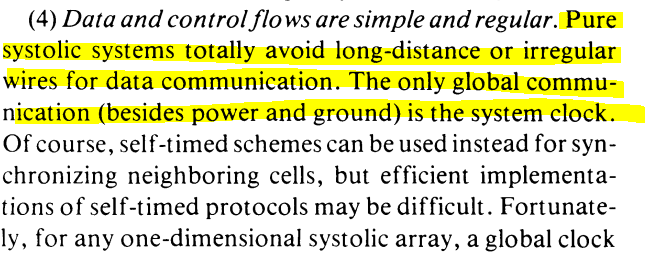
***The three most common resolutions are standard high definition (called HD Ready or 720p) which is 1280 x 720 pixels, full high definition (FHD, 1080p) which measures 1920 x 1080 pixels, and ultra-high definition (UHD, 4K) which has a resolution of 3,840 by 2160 pixels.***

Ventaja del systolic array

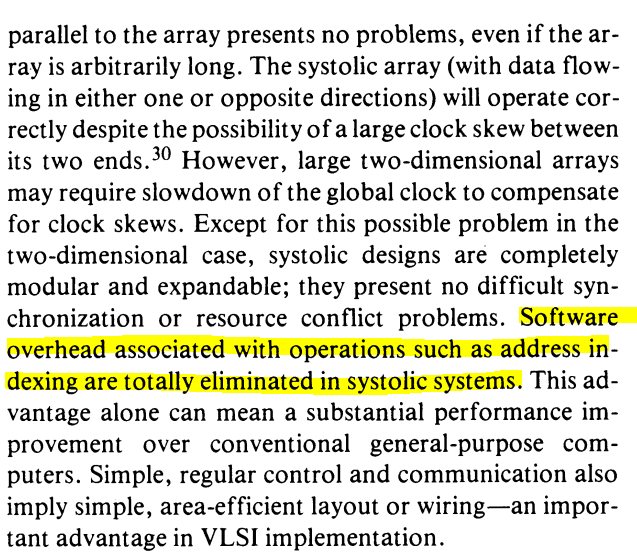
* Data and control flows are simple and regular

Los sistemas puramente sistólicos evitan a toda costa conexiones a larga distancia o irregulares para la comunicación de datos.

La única comunicación global (además del power y ground) es el reloj del sistema.



* ***Ventaja sobre un general-purpose computer***
  + ***Te olvidas del overhead provocado por el address indexing***



Lógica secuencial

En diseño digital, la lógica secuencial es un tipo de circuito lógico en donde sus salidas no solo dependen del valor actual de sus entradas, sino también de la secuencia de entradas anteriores, es decir del historial de sus entradas.

Lógica combinacional

Sus salidas solo dependen de los valores de entrada presente.

Básicamente, los circuitos secuenciales tienen estados (memoria) mientras que los combinacionales no.

Circuito síncrono

El estado del circuito cambia solo *at discrete times* en respuesta a una señal de reloj.

**RESUMEN DE POR QUÉ UTILIZAR SYSTOLIC ARRAYS**

* **Simple**
* **Modular and expandable**
* **High performance**

**From the user point of view, it’s easy to use, she pumps in the input data and then receives the results on the fly or at the end of the computation.**

