

Válaszok Dr. Czúni László „CNN-alapú képfeldolgozó és adaptív optikai rendszer FPGA-s implementációi” című disszertációmmal kapcsolatos észrevételeire, javaslataira.

Tisztelt Dr. Czúni László!

Mindenekelőtt szeretném megköszönni a disszertációmban, valamint a tézisfüzetemben foglaltakkal kapcsolatos építő jellegű észrevételeit és javaslatait. Az alábbiakban válaszolni szeretnék a felmerült kérdésekre, észrevételekre.

Ennél fontosabb hiányosság, hogy a képfeldolgozás céljából készült rendszert leíró dolgozat egyetlen egy képet sem mutat be a rendszer által feldolgozandó és feldolgozott képekről.

A disszertációmban elsősorban architektúrális kérdésekkel foglalkozom, és ezen architektúrákon belül is a különböző algoritmusok FPGA alapú megvalósíthatóságával.

CNN optikai megvalósítása során miért nehéz megoldani az optikai visszacsatolást?

Optikai-elektromos-optikai átalakítást kell elvégezni, amely negatívan hathat a rendszer sebességére és pontosságára. A CNN optikai megvalósításában (POAC) ezért nem is alkalmaznak visszacsatolást csak előreccsatolást.

Mi az optikai rendszer működésének fizikai elve?

A dolgozatomban említett optikai CNN megvalósítás a POAC egy teljes mértékben optikai elven működő rendszer, melyben különböző prizmák, sugárosztók, félig áteresztő szűrők és optikai korrelátorok találhatóak. A POAC rendszer egy olyan optikai CNN-t valósít meg, amelyben csak előreccsatolás van. Ebben a megvalósításban könnyen megoldható a template-ek méretének tetszőleges növelése, valamint a hely-variáns template-ek alkalmazása, mely az analóg vagy emulált digitális CNN-UM megvalósításban, csak nagyobb áldozatok árán valósítható meg.

Mi a különbség a nagyteljesítményű logika és a nagy számítási teljesítmény között?

A mondat helyesen megfogalmazva: A Virtex-6 FPGA LXT alcsaládjában a legnagyobb az általános logikai erőforrások száma (LUT), míg az SXT alcsaládba tartozó FPGA esetében pedig a rendelkezésre álló DSP Slice-ok száma a legnagyobb.

Hogyan célszerű megoldani olyan eseteket, amikor valamilyen hullámfüggvény, exponenciális, logaritmusos függvény, vagy egyéb nemlinearitás a feladat?

Egyrészt ki lehet terjeszteni az elkészült architektúra nemlineáris template memóriáját annak érdekében, hogy tetszőleges n -ed fokú polinomokból álljon össze a nemlinearitás. Ekkor egy Taylor sorfejtés után szakaszonként közelíthető a nemlinearitás. A belső RAM memóriában ekkor a Taylor-sor együtthatóit kell eltárolni. Másrészt a rendszer eléggé moduláris ahhoz, hogy az így eltárolt adatok alapján egy tetszőleges paraméterezhető függvény felhasználásával határozzuk meg a nemlineáris template értékeit. A napjainkban rendelkezésre álló C alapú fejlesztőeszközök lehetővé teszik ennek a módosításnak a gyors kivitelezését.

Az SAD, ill. sorozatos SAD kiértékelések egy adott környezetben alapvetően egyszerű feladatok, igen sok helyen használják őket a képfeldolgozásban. Éppen ezért számtalan gyorsítása is létezik, pl. hierarchikus keresés, szomszédos területek eredményének felhasználása (lehetséges korreláció miatt), dinamikus programozás (amikor az SAD számítását abba lehet hagyni, ha egy jobb megoldást korábban már találtunk), ill. utóbbi kettő kombinációja. Sajnos a dolgozat ezeket az elvi lehetőségeket nem tárgyalja. Célszerű lett volna egy szimulátorral a probléma (HS korrekció) szempontjából leghatékonyabb SAD algoritmus meghatározni, majd a kiválasztott algoritmust implementálni FPGA-ra.

Köszönöm az észrevételt azonban a rendszer működésének sajátosságai miatt a képérzékelő szenzorról érkező adatokat a rendelkezésre állásuk pillanatában már fel kell dolgozni. A valós idejű és kis késleltetésű számítás elérése érdekében nincs lehetőség a teljes kép eltárolására.

Mivel az MPEG kódolásban is gyakran használnak ilyen, ill. hasonló módszereket (pl. korrelációs számítások), ezért nyilvánvalóan több FPGA-s implementációnak is kell léteznie a problémára. A dolgozat 88. oldalán a szerző elveti ezen összehasonlításokat (a [42], [48], [49] cikkekkel), arra hivatkozva, hogy azok Altera és nem az általa használt Xilinx áramkörökre lettek kidolgozva. Ezzel a megállapítással nem teljesen értek egyet, hiszen a kétféle chip az alap algoritmus szempontjából lényegesen nem különbözhet, ha igen, ezt meg kellene indokolni.

Az Altera és Xilinx cégek által forgalmazott FPGA-k architektúráis különbségei miatt csak a végrehajtási idők hasonlíthatók össze objektíven, az architektúra felület igénye nem, így pont a kialakított architektúrák hatékonysága nem lesz összevethető az FI paraméter segítségével.

Kérdés továbbá, hogy a 84. oldalon az erőforrás igény elvi becslés vagy pedig implementáció utáni mérés eredménye?

A Xilinx ISE Design Suite programcsomagban található szintézis eszközök az architektúra szintézise és implementációja során létrehoznak egy úgynevezett Map Report file-t, mely tartalmazza az implementált architektúra általános és dedikált erőforrásigényét. A 84. oldalon feltüntetett erőforrás igény eredmények, tehát implementáció utáni mérésekből származnak.

A 4-4. táblázatban melyik kártyára értendők az adatok az SAD esetében?

Az összehasonlításban szereplő hagyományos korreláció alapú hullámfront szenzor megvalósítása egy Xilinx Virtex-4 SX35-10 FPGA-n valósult meg. Az általam létrehozott SAD alapú hullámfront szenzor architektúra pedig az FPGA alapú adaptív optikai rendszer magját képező Xilinx Spartan-3 XC3S4000 FPGA-n. Hogy az összehasonlítás helyes legyen az általam létrehozott architektúra működési sebességnek nem a maximum 120MHz-t, hanem a korreláció alapú rendszer által elérhető 100MHz sebességet választottam.

A 3.3.3 fejezetben a tesztek 128x128 méretű képekre vonatkoznak. Lényegesen nagyobb kép esetén mi történne?

Az architektúrával elérhető számítási sebesség az adott FPGA típuson implementálható nullad- vagy elsőrendű FALCON processzorok számától függ. Az implementálható processzorok száma pedig az állapotérték bitszélességének függvénye. A képméret a kép egy sorának eltárolásához szükséges BRAM méretet határozza meg. Mivel a Xilinx Virtex-4 FPGA családig egy BRAM 18Kbit, akkor 18 bites

bemeneti pixel bitszélességgel számolva 1Kx1K méretű képeket tudunk kezelni egy processzorral. A Virtex-5 FPGA családtól pedig a rendelkezésre álló 36Kbit méretű BRAM-ok segítségével 2Kx2K méretű képeket is kezelhetünk. Mivel azonban megvan annak a lehetősége, hogy két 36Kbit BRAM-ot egy 72Kbit BRAM memóriává fűzzünk össze, így akár 4Kx4K méretű képeket is tudunk kezelni. Természetesen ekkor egy-egy kép feldolgozásának ideje a pixelek számával arányosan fog növekedni.

Felmerül a kérdés az algoritmusok helyességét illetően. Honnan tudhatjuk, hogy valamilyen tervezési hiba („bug”) nem rontja-e el a futások eredményét, akár kismértékben is? Van-e összehasonlítási alap?

Az elkészített architektúra verifikációja FPGA szimulációs környezetben egy előre meghatározott teszt minta segítségével történt. A verifikáció lényege, az architektúrában található adatutak, valamint számítási egységek helyes működésének ellenőrzése volt. A tesztminta minden esetben egy olyan kép volt, amelynek ha a pixeleit sorba rakjuk, akkor minden pixel értéke eggyel nagyobb, mint az azt megelőző pixel értéke.

Végezetül még egyszer szeretném megköszönni bírálóm Dr. Czúni László észrevételeit, javaslatait, valamint a munkám pozitív értékelését.

Szeged, 2013.05.15



Kincses Zoltán

egyetemi tanársegéd

SZTE TTIK ITCS MIT