Prepoznavanje objekata na RISC-V arhitekturi računala

Budimir-Bekan, Borna

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

DIPLOMSKI RAD br. 594

PREPOZNAVANJE OBJEKATA NA RISC-V ARHITEKTURI RAČUNALA

Borna Budimir-Bekan

Zagreb, lipanj 2024.

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

DIPLOMSKI RAD br. 594

PREPOZNAVANJE OBJEKATA NA RISC-V ARHITEKTURI RAČUNALA

Borna Budimir-Bekan

Zagreb, lipanj 2024.

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

Zagreb, 4. ožujka 2024.

DIPLOMSKI ZADATAK br. 594

Pristupnik:	**Borna Budimir-Bekan (0036523784)**
Studij:	Računarstvo
Profil:	Znanost o podacima
Mentor:	izv. prof. dr. sc. Vladimir Čeperić

Zadatak: **Prepoznavanje objekata na RISC-V arhitekturi računala**

Opis zadatka:

U ovom diplomskom radu istražit će se i implementirati tehnike prepoznavanja objekata na RISC-V platformi, s posebnim fokusom na razvojnu platformu StarFive VisionFive 2 koja sadrži integrirani GPU. Rad će obuhvatiti razvoj, optimizaciju i implementaciju algoritama za detekciju objekata koristeći mogućnosti GPU ubrzanja putem OpenCL tehnologije. Cilj je maksimalno iskoristiti hardverske resurse VisionFive 2 platforme, uključujući CPU i integrirani GPU, kako bi se postiglo efikasno i brzo procesiranje slika i video materijala. Detaljna analiza i usporedba performansi različitih modela dubokog učenja poput MobileNet SSD i YOLO provest će se unutar RISC-V okruženja. Eksperimentalni dio rada uključivat će testiranje algoritama na skupovima podataka kao što su MNIST i CIFAR-10. U radu će se priložiti izvorni programski kodovi, analiza dobivenih rezultata te će se detaljno objasniti korištene metode i literatura.

Rok za predaju rada: 28. lipnja 2024.

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

DIPLOMSKI RAD br. 594

PREPOZNAVANJE OBJEKATA NA RISC-V ARHITEKTURI RAČUNALA

Borna Budimir-Bekan

Zagreb, veljača, 2025.

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

Zagreb, 4. ožujka 2024.

DIPLOMSKI ZADATAK br. 594

Pristupnik:	**Borna Budimir-Bekan (0036523784)**
Studij:	Računarstvo
Profil:	Znanost o podacima
Mentor:	izv. prof. dr. sc. Vladimir Čeperić

Zadatak: **Prepoznavanje objekata na RISC-V arhitekturi računala**

Opis zadatka:

U ovom diplomskom radu istražit će se i implementirati tehnike prepoznavanja objekata na RISC-V platformi, s posebnim fokusom na razvojnu platformu StarFive VisionFive 2 koja sadrži integrirani GPU. Rad će obuhvatiti razvoj, optimizaciju i implementaciju algoritama za detekciju objekata koristeći mogućnosti GPU ubrzanja putem OpenCL tehnologije. Cilj je maksimalno iskoristiti hardverske resurse VisionFive 2 platforme, uključujući CPU i integrirani GPU, kako bi se postiglo efikasno i brzo procesiranje slika i video materijala. Detaljna analiza i usporedba performansi različitih modela dubokog učenja poput MobileNet SSD i YOLO provest će se unutar RISC-V okruženja. Eksperimentalni dio rada uključivat će testiranje algoritama na skupovima podataka kao što su MNIST i CIFAR-10. U radu će se priložiti izvorni programski kodovi, analiza dobivenih rezultata te će se detaljno objasniti korištene metode i literatura.

Rok za predaju rada: 28. lipnja 2024.

Prije svega želim zahvaliti svom mentoru, prof. dr. sc. Vladimiru Čeperiću. Hvala Vam na iznimnom strpljenju i razumijevanju tijekom preddiplomskog i diplomskog studija. Učinili ste mi studij istovremeno izazovnijim i zabavnijim zadavanjem kreativnih zadataka koji su bili van moje sfere razmišljanja. Bez Vas ih se vjerojatno ne bih nikada ni dotakao i naučio nove i interesantne stvari. Hvala Vam na svemu, boljeg mentora nisam mogao imati.

Mama i tata, kroz svaki trenutak mog života ste istovremeno bili uz mene ali mi i dopuštali da radim stvari na svoj način. Oblikovali ste me u čovjeka koji jesam i kojime mogu reći da se ponosim. Bez Vas ne bih bio ovdje gdje jesam i na tome ću Vam biti vječito zahvalan. Volim vas i nadam se da ste ponosni.

Baka Alma i deda Maleni, vas dvoje ste bili i ostajete biti sastavni dio mog života. Hvala vam na beskrajnoj ljubavi koju ste mi pružali tijekom mog odrastanja i mnogim vrijednostima koje ste me naučili. Voli vas vaš unuk.

Antonija i Laura, hvala vam na predivnom odnosu koji imamo. Nadam se da u vašim glavama, nakon ovog diplomskog, više neću imati 16 godina iako ću uvijek biti vaš mali brat. Bolje sestre ni sam nisam mogao izabrati. Voli vas vaš bratec.

Lana, Leona, Mauro i Tonka, nastojat ću biti uz vas u svakom dijelu vašeg životnog puta i pomagati vam koliko znam i umijem. Nek' vam ovo bude dokaz da je u životu sve moguće. Voli vas vaš ujo.

Posebne zahvale idu mojoj divnoj djevojci Mariji, hvala ti što si me bodrila tijekom čitavog trajanja ovog rada. Tvoja konstantna motivacija i pokoji ljuti pogled na moju neiscrpnu prokrastinaciju su me pogurali da nastavim raditi i pisati čak i u trenutcima kad sam mentalno odustao. Volim te i oprosti što sam te izludio.

Svoj rodbini i prijateljima koje nisam ranije spomenuo – jedno veliko hvala!

Sadržaj

1. Uvod

U posljednjih nekoliko godina, područje računalnog vida i detekcije objekata doživjelo je značajan napredak, prvenstveno zahvaljujući razvoju dubokih neuronskih mreža i napretku računalnog hardvera. Istovremeno, RISC-V arhitektura isplivava kao otvorena i fleksibilna alternativa tradicionalnim procesorskim arhitekturama, privlačeći sve veću pažnju akademske zajednice i industrije. Ovaj rad istražuje mogućnosti implementacije naprednih algoritama za detekciju objekata na RISC-V platformi, s posebnim naglaskom na iskorištavanje grafičkog procesora za ubrzanje računski zahtjevnih operacija.

Detekcija objekata predstavlja jedan od temeljnih problema u području računalnog vida, s širokim spektrom primjena - od autonomnih vozila i robotike do sustava videonadzora i kontrole kvalitete u industriji. Tradicionalni pristupi rješavanju ovog problema često su računski zahtjevni i teško primjenjivi u sustavima s ograničenim resursima. Međutim, razvoj optimiziranih arhitektura neuronskih mreža poput MobileNet SSD i YOLO otvorio je mogućnost efikasne implementacije ovih sustava na embedded platformama.

VisionFive 2 kineske kompanije StarFive, kao moderna RISC-V razvojna platforma s integriranim GPU-om, predstavlja idealan sustav za istraživanje mogućnosti implementacije algoritama za detekciju objekata. Posebnost ove platforme leži u činjenici da kombinira prednosti otvorene RISC-V arhitekture s mogućnostima hardverskog ubrzanja putem GPU-a, što otvara prostor za značajne optimizacije u području obrade slika i video sadržaja.

Glavni ciljevi ovog diplomskog rada su implementacija i optimizacija algoritama za detekciju objekata na RISC-V arhitekturi, razvoj efikasnih metoda za iskorištavanje GPU akceleracije putem OpenCL tehnologije, evaluacija performansi različitih modela dubo-

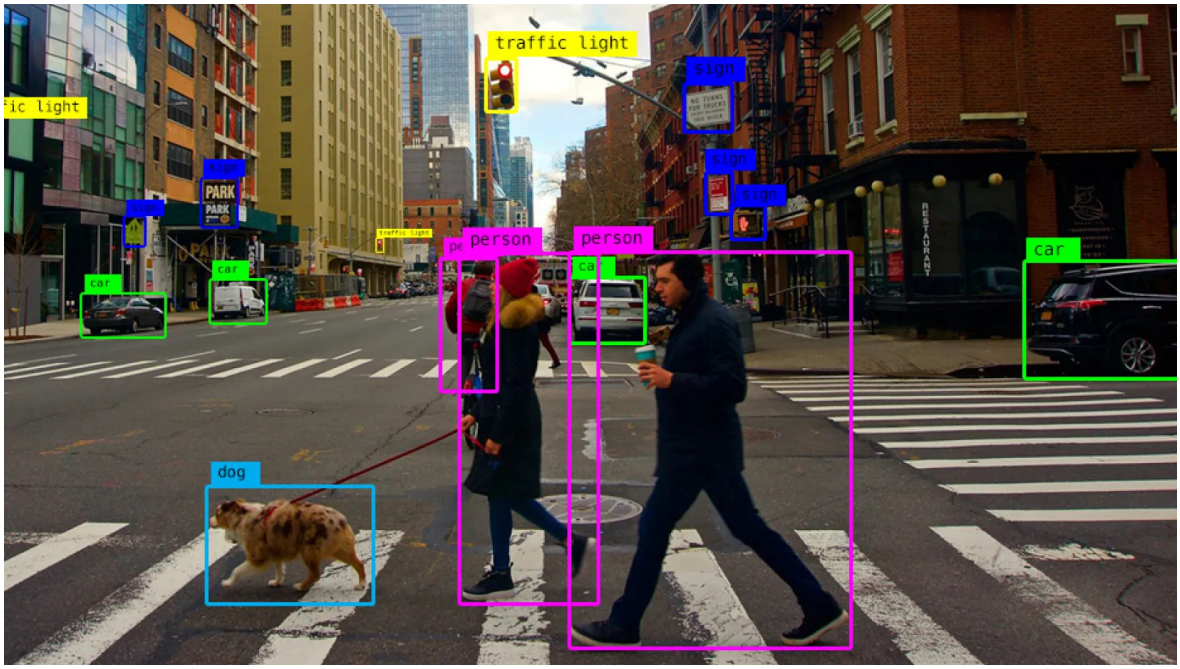
kog učenja u kontekstu embedded sustava te analiza mogućnosti i ograničenja StarFive VisionFive 2 platforme u aplikacijama računalnog vida.

Očekivani znanstveni doprinosi ovog rada uključuju razvoj optimizirane implementacije algoritama za detekciju objekata prilagođene RISC-V arhitekturi, analizu performansi i mogućnosti GPU akceleracije na VisionFive 2 platformi, komparativnu analizu različitih pristupa implementaciji dubokog učenja na embedded sustavima te praktične smjernice za razvoj aplikacija računalnog vida na RISC-V platformama.

Nakon uvodnog poglavlja, u drugom poglavlju predstavljena je teorijska podloga potrebna za razumijevanje implementiranih rješenja, uključujući pregled RISC-V arhitekture, metoda detekcije objekata i principa GPU akceleracije. Treće poglavlje opisuje razvojnu okolinu i korištene alate, dok četvrto poglavlje detaljno predstavlja implementaciju sustava. U petom poglavlju prikazani su rezultati eksperimentalne evaluacije implementiranih rješenja. Rad završava zaključkom koji sumira postignute rezultate i daje smjernice za budući rad u ovom području.

Za praktičnu evaluaciju razvijenih rješenja korišteni su standardni skupovi podataka MNIST i CIFAR-10, koji omogućuju objektivnu usporedbu s postojećim rješenjima u literaturi. Posebna pažnja posvećena je optimizaciji performansi i analizi mogućnosti praktične primjene razvijenih rješenja u realnim uvjetima.

2. Prepoznavanje objekata u računalnom vidu



Slika 2.1. Primjer prepoznavanja objekata na fotografiji

Prepoznavanje objekata u računalnom vidu predstavlja jedan od ključnih problema u području umjetne inteligencije i obrade slike. Cilj prepoznavanja objekata je identificirati i locirati objekte unutar slike ili videozapisa, pri čemu se koriste različite metode strojnog učenja i dubokih neuronskih mreža. Ova tehnika nalazi široku primjenu u autonomnim vozilima, sigurnosnim sustavima, industrijskoj automatizaciji i mnogim drugim područjima.

2.1. Opći pristup prepoznavanju objekata

Tradicionalni pristupi prepoznavanju objekata oslanjali su se na ručno dizajnirane značajke poput SIFT-a (Scale-Invariant Feature Transform)[1] i HOG-a (Histogram of Oriented Gradients), ali su takvi pristupi pokazali ograničenu učinkovitost u složenim scenarijima. S razvojem dubokog učenja, konvolucijske neuronske mreže (CNN) postale su dominantna metoda za prepoznavanje objekata.

Moderni pristupi prepoznavanju objekata obično se dijele na dvije glavne kategorije:

- **Dvostupanjski modeli** – primjer su R-CNN i njegove varijacije (Fast R-CNN, Faster R-CNN). Ovi modeli prvo generiraju prijedloge regija (region proposals) te potom klasificiraju objekte unutar tih regija.
- **Jednostupanjski modeli** – poput YOLO (You Only Look Once) i SSD (Single Shot MultiBox Detector), koji predviđaju položaj i klasu objekata u jednoj obradi slike, čime postižu veću brzinu.

2.2. YOLO obitelj modela

YOLO (You Only Look Once)[2] je serija jednostupanjskih modela za prepoznavanje objekata koji su dizajnirani za brzu i točnu detekciju. Umjesto segmentacije slike u regije interesa, YOLO dijeli sliku na mrežu i predviđa pravokutnike (bounding boxes) te pridružene klase objekata u jednoj prolaznoj obradi.

Različite verzije YOLO modela donijele su poboljšanja u preciznosti i brzini:

- **YOLOv1** – prvi model koji je uveo koncept prepoznavanja objekata u jednom prolazu.
- **YOLOv3 i YOLOv4** – značajna poboljšanja u preciznosti i učinkovitosti kroz optimizacije mrežne arhitekture.
- **YOLOv5** – poboljšanja u performansama, prilagođenost različitim veličinama modela (n, s, m, l, x).
- **YOLOv11** – najnovija verzija s optimizacijama u arhitekturi i boljoj generalizaciji.

YOLO modeli su često korišteni u aplikacijama gdje je brzina presudna, kao što su autonomna vozila i sustavi za nadzor.

2.3. MobileNet SSD obitelj modela

MobileNet SSD (Single Shot MultiBox Detector)[3] je serija modela dizajniranih za prepoznavanje objekata na uređajima s ograničenim računalnim resursima. Ovi modeli koriste MobileNet kao baznu mrežu, koja je optimizirana za brzinu i malu potrošnju memorije.

SSD model[3] radi na principu višestrukih mapa značajki (feature maps) kako bi otkrio objekte različitih veličina. Kombiniranjem s MobileNet arhitekturom postiže dobar omjer između točnosti i brzine obrade, čineći ga pogodnim za mobilne i ugrađene sustave.

Prednosti MobileNet SSD modela uključuju:

- Male računalne zahtjeve u usporedbi s dubljim modelima poput Faster R-CNN.
- Brzu inferenciju, prikladnu za real-time aplikacije.
- Fleksibilnost u implementaciji na mobilnim uređajima i IoT platformama.

2.4. Zaključak

Prepoznavanje objekata predstavlja jedno od ključnih područja računalnog vida, a duboko učenje omogućilo je značajan napredak u preciznosti i brzini obrade. YOLO modeli pružaju rješenja visokih performansi pogodna za primjene u stvarnom vremenu, dok MobileNet SSD omogućava prepoznavanje objekata na uređajima s ograničenim resursima. Odabir odgovarajućeg modela ovisi o specifičnim zahtjevima aplikacije, uključujući točnost, brzinu i dostupne računalne resurse. U svrhu ovog rada, obje obitelji modela će biti detaljno testirane i uspoređene.

3. RISC-V

3.1. Motivacija

RISC-V je arhitektura procesora otvorenog koda, temeljena na paradigmi Reduced Instruction Set Computing (RISC). Njezin razvoj započeo je u svibnju 2010. godine na Sveučilištu Berkeley u Kaliforniji pod vodstvom profesora Krste Asanovića, uz sudjelovanje njegovih studenata diplomskog studija kao dio istraživačkog projekta Parallel Computing Laboratory (Par Lab). Primarni cilj projekta bio je istražiti nove pristupe paralelnom računarstvu, s posebnim naglaskom na efikasne i skalabilne procesorske arhitekture.

Projekt je financiran od strane tehnoloških kompanija poput Intela i Microsofta te institucija kao što su DARPA i savezna država Kalifornija. Baš kao i svi ostali projekti Par Lab-a, RISC-V je od početka razvijan kao arhitektura otvorenog koda, što znači da je njegova dokumentacija slobodno dostupna široj javnosti. Ova filozofija razlikuje RISC-V od dominantnih arhitektura poput x86 (Intel, AMD) i ARM (ARM Holdings), koje su zatvorene i zahtijevaju plaćanje licenci za korištenje. Prvi priručnik za arhitekturu[4] objavljen je 2011. godine, a autori su, uz Krstu Asanovića, i njegovi studenti Yunsup Lee i Andrew Waterman.

Dostupnost i otvorenost arhitekture ključni su faktori koji su doprinijeli brzom rastu popularnosti RISC-V-a. Ovakav pristup omogućuje bilo kojem pojedincu ili tvrtki da razvija vlastita procesorska rješenja bez dodatnih troškova licenci, čime se otvara prostor za inovacije i prilagodbu arhitekture specifičnim potrebama. Danas je RISC-V pod nadzorom organizacije RISC-V International, koja je osnovana 2015. godine i okuplja preko 400 članova, uključujući akademske institucije, startupe i velike tehnološke kompanije. Fakultet elektrotehnike i računarstva u Zagrebu također je član organizacije, što omogućuje pristup najnovijim razvojnim alatima i istraživačkim resursima.

3.2. Povijest

Arhitektura RISC-V nije nastala izolirano, već je rezultat dugogodišnjeg istraživanja i razvoja unutar područja RISC procesora [5]. Prije RISC-V-a, na Sveučilištu Berkeley razvijeno je nekoliko generacija RISC arhitektura koje su postavile temelje za moderni dizajn procesora.

3.2.1. Prethodni RISC projekti s Berkeleyja

Naziv RISC-V označava petu iteraciju arhitekture razvijene na Berkeleyju, a prethodne verzije uključuju:

- RISC-I (1981.) – prvi funkcionalni RISC procesor razvijen na Berkeleyju. Imao je svega 31 instrukciju i koristio je jednostavan dizajn koji je omogućio visoku efikasnost izvršavanja.
- RISC-II (1983.) – poboljšana verzija RISC-I procesora, optimizirana za bolju performansu i prošireni skup instrukcija.
- SOAR (1984.) – poznat i kao RISC-III, ovaj projekt bio je usmjeren na eksperimentiranje s arhitekturom optimiziranom za operacijske sustave. Implementacija je obuhvaćala podršku za naprednije upravljanje memorijom i bolje upravljanje resursima procesora.
- SPUR (1988.) – eksperimentalni projekt, poznat i kao RISC-IV, koji je uveo napredne značajke poput podrške za virtualnu memoriju i multiprocesorsko izvođenje zadataka.

3.2.2. Razvoj RISC-V arhitekture

Nakon ovih ranih istraživanja, RISC arhitektura postala je osnova za mnoge moderne procesorske dizajne. Primjerice, arhitektura ARM, koja danas dominira tržištem mobilnih uređaja, temelji se na RISC principima. Međutim, do pojave RISC-V-a nije postojala univerzalna i slobodno dostupna RISC arhitektura koja bi omogućila prilagodbu i razvoj bez ograničenja licenciranja.

U razdoblju između 2010. i 2015. godine, razvoj RISC-V arhitekture na Berkeleyu

prošao je kroz nekoliko faza. Prvi konkretni implementacijski dizajni razvijeni su kao dio akademskih istraživanja, no ubrzo su prepoznati i od strane industrije. Godine 2015. osnovana je organizacija RISC-V International kako bi upravljala standardizacijom i širenjem ekosustava oko ove arhitekture.

3.2.3. Trenutačno stanje RISC-V arhitekture

Danas je RISC-V jedna od najbrže rastućih procesorskih arhitektura u svijetu. Sve veći broj tvrtki i istraživačkih institucija ulaže u razvoj procesora temeljenih na RISC-V-u, a arhitektura se primjenjuje u širokom spektru uređaja, od ugrađenih sustava i mikrokontrolera do serverskih procesora i superračunala.

Neke od ključnih prednosti koje doprinose širenju RISC-V arhitekture uključuju:

- Modularni dizajn – omogućuje proizvođačima da prilagode skup instrukcija specifičnim primjenama, čime se smanjuje kompleksnost i povećava energetska efikasnost.
- Otvoreni standard – uklanja potrebu za plaćanjem licenci, što omogućuje brži razvoj i niže troškove proizvodnje.
- Aktivna zajednica – tisuće inženjera i istraživača kontinuirano rade na optimizaciji arhitekture i razvoju alata za programiranje i simulaciju.

U posljednjih nekoliko godina, RISC-V je privukao interes velikih tehnoloških kompanija poput Western Digitala, Nvidije, SiFive-a i Intela, koje aktivno razvijaju vlastite implementacije. Također, sve veći broj operacijskih sustava, uključujući Linux distribucije poput Debian-a i Fedora-e, dobiva punu podršku za RISC-V procesore, što dodatno ubrzava prihvaćanje arhitekture na tržištu.

3.3. Budućnost RISC-V arhitekture

Budućnost RISC-V arhitekture izgleda obećavajuće, s obzirom na sve veće usvajanje u različitim sektorima industrije. Očekuje se da će RISC-V u narednim godinama postati ozbiljan konkurent ARM i x86 arhitekturama, osobito u segmentima ugrađenih sustava, IoT uređaja i HPC (High-Performance Computing) rješenja.

Ključni izazovi s kojima se RISC-V ekosustav trenutačno suočava uključuju potrebu za boljom podrškom za softver i operacijske sustave, optimizaciju performansi u visoko zahtjevnim primjenama te daljnju standardizaciju proširenja instrukcijskog skupa. No, s obzirom na rastući interes akademske i industrijske zajednice, RISC-V ima potencijal postati dominantna otvorena procesorska arhitektura u budućnosti.

4. Odabir RISC-V jednopločnog računala

Za potrebe rada tražili smo jednopločno računalo (eng. single board computer, SBC) idućih karakteristika:

- **RISC-V arhitektura procesora** – kako bi istražili mogućnosti i performanse modernih otvorenih arhitektura u kontekstu računalnog vida.
- **Integrirani GPU** – potreban za hardversku akceleraciju u zadacima računalnog vida i strojne obrade slike.
- **RAM: 8 GB ili više** – osigurava dovoljno memorije za rad s modelima dubokog učenja i obradu slike u stvarnom vremenu.
- **I/O: mogućnost spajanja web kamere** – nužno za izvođenje algoritama na prijenosu uživo.
- **Povezivost: Ethernet port ili podrška za USB Wi-Fi adapter** – omogućava daljinsku komunikaciju i prijenos podataka s uređaja.
- **OS: dostupna distribucija Linuxa** – kako bi se osigurao jednostavan razvoj i integracija s postojećim alatima otvorenog koda.
- **Dobar omjer cijene i očekivanih performansi** – uređaj treba biti isplativ u usporedbi s performansama koje nudi.
- **Postojeća dokumentacija i podrška** – olakšava razvoj i rješavanje potencijalnih problema.
- **Dostupnost u EU** – mnogi dostupni RISC-V SBC-ovi dolaze iz Kine, što može otežati nabavu i podršku.

4.1. Dostupni SBC-ovi u trenutku odabira

U trenutku odabira (travanj 2024.) na tržištu su bili dostupni sljedeći RISC-V SBC-ovi koji su zadovoljavali barem neke od kriterija:

- **CanMV-K230** – kompaktni SBC baziran na Kendryte K230 procesoru, dizajniran za low-power edge AI aplikacije. Ima ugrađeni Neural Processing Unit (NPU) optimiziran za zadatke prepoznavanja objekata, ali je ograničen sa samo 64 MB RAM-a, što ga čini neprikladnim za složenije modele računalnog vida. Osim toga, nema standardni GPU već koristi hardversku akceleraciju putem NPU-a.

- **LicheePi 4A** – jedan od najnaprednijih RISC-V SBC-ova, temeljen na T-Head TH1520 procesoru. Dolazi s integriranim GPU-om i 16 GB RAM-a, što ga čini snažnim kandidatom za zadatke računalnog vida. Također ima četiri RISC-V jezgre i podršku za PCIe 3.0, što omogućava proširenje s dodatnim akceleratorima. No, uređaj je u trenutku analize imao manje razvijenu softversku podršku i nestabilne GPU drivere.

- **StarFive VisionFive 2** – jedini od navedenih SBC-ova s službenom podrškom za OpenCL na PowerVR BXE-4-32 GPU-u, što omogućava efikasno iskorištavanje GPU-a u zadacima računalnog vida. Ima 8 GB RAM-a, četverojezgreni RISC-V procesor i dostupne Linux distribucije poput Debian-a i Fedora-e. Također je široko podržan u zajednici, s redovnim ažuriranjima softvera.

4.2. Odabir VisionFive 2

Iako su **CanMV-K230** i **LicheePi 4A** na papiru i prema testovima ostvarivali nešto bolje rezultate u zadacima prepoznavanja objekata, VisionFive 2 se pokazao najboljim izborom za ovaj projekt iz nekoliko ključnih razloga:

- Ispunjava sve postavljene tehničke zahtjeve, uključujući dedicirani GPU i dostupne OpenCL drivere.

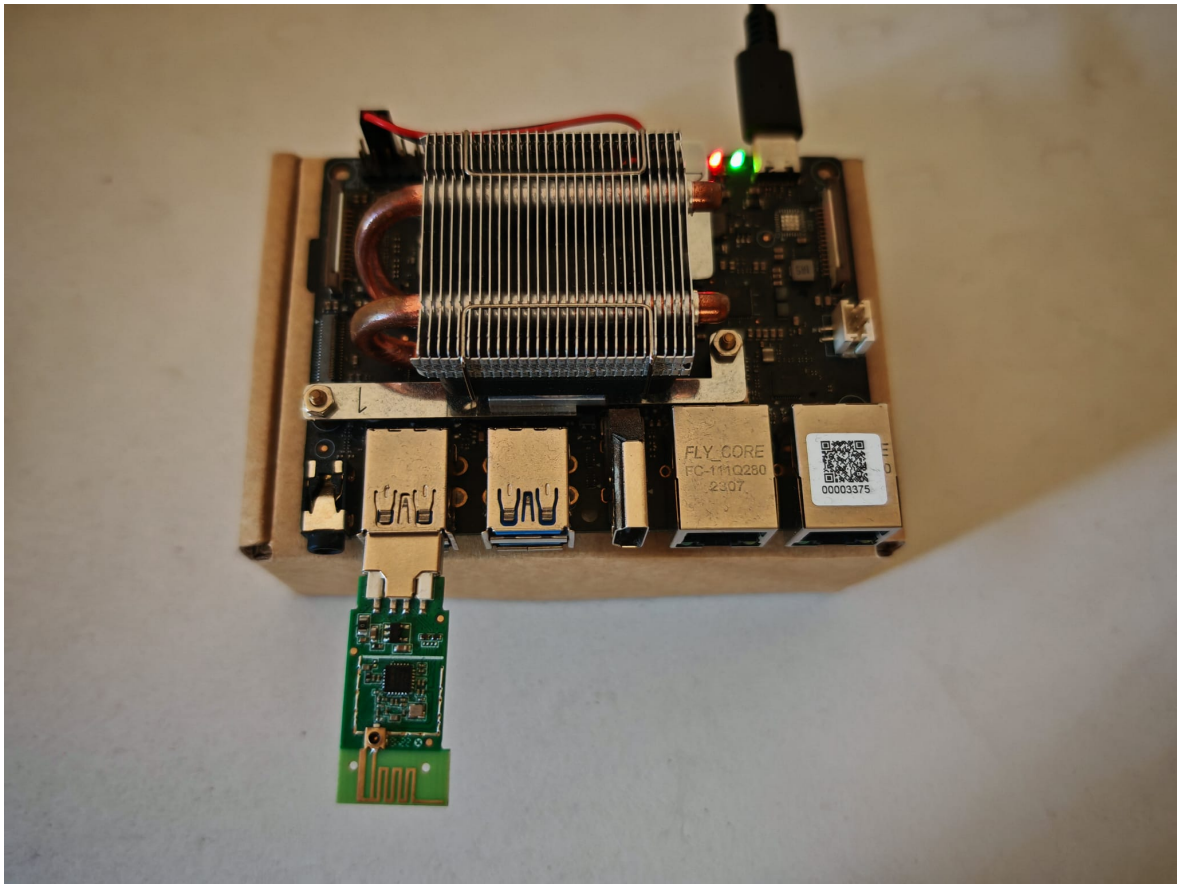
- Ima stabilnu softversku podršku, uključujući redovita ažuriranja i aktivnu zajednicu.

- Cijena uređaja bila je povoljna – cca **€120 u trenutku kupovine**, što ga čini znatno isplativijim u usporedbi s LicheePi 4A.
- Dokumentacija je detaljna i dostupna na engleskom jeziku, uz podršku u obliku aktivnog bloga i foruma.

Na temelju ovih faktora, VisionFive 2 je odabran kao najbolje rješenje za razvoj računalnog vida na RISC-V arhitekturi unutar ovog projektnog zadatka.

5. StarFive VisionFive 2

VisionFive 2 je druga iteracija RISC-V SBC-a kineske kompanije StarFive koja napominje kako je to prvi SBC te vrste s integriranim GPU-om koji bi u teoriji trebao značajno smanjiti vrijeme inferencije u zadacima prepoznavanja objekata.



Slika 5.1. StarFive VisionFive 2

5.1. Specifikacije

Iduća tablica sadrži tehničke specifikacije računala[6].

Tablica 5.1. Specifikacije VisionFive 2

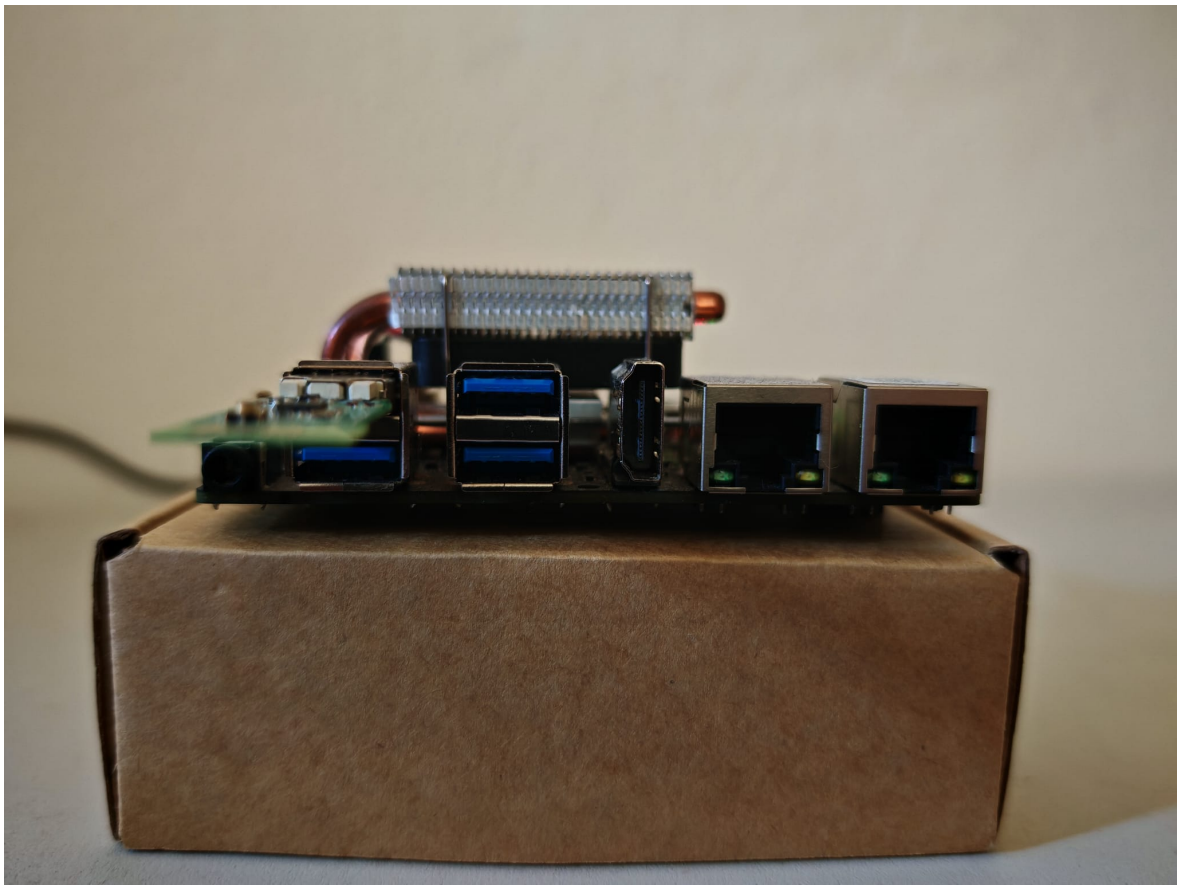
Komponenta	Naziv	Opis
Procesor	StarFive JH-7110	StarFive JH-7110 s RISC-V četverojezgrenim CPU-om s 2 MB L2 predmemorije i nadzornom jezgrom, podržava RV64GC ISA, rad do 1.5 GHz
	Imagination GPU	IMG BXE-4-32 MC1 s radnom frekvencijom do 600 MHz
Memorija		2 GB/4 GB/8 GB LPDDR4 SDRAM, do 2800 Mbps
Pohrana	TF kartica	Mogućnost pokretanja sustava s TF kartice
	Flash	Firmware za U-Boot i BootLoader
Multimedija	Video izlaz	1 × 2-linijski MIPI DSI, do 1080p@30fps; 1 × 4-linijski MIPI DSI, do 2K@30fps; 1 × HDMI 2.0, do 4K@30fps ili 2K@60fps
	Kamera	1 × 2-linijski MIPI CSI, do 1080p@30fps
	Koder/Dekoder	Video dekoder do 4K@60fps (H264/H265), video koder do 1080p@30fps (H265), JPEG koder/dekoder
	Audio	4-polni stereo audio priključak
Povezivost	Ethernet	2 × RJ45 gigabitna Ethernet priključka
	USB	4 × USB 3.0 priključka, 1 × USB uređaj preko USB-C
	M.2 Priključak	M.2 M-Key
	eMMC Utori	eMMC modul za OS i podatkovnu pohranu

nastavak na sljedećoj stranici

Tablica 5.1. (nastavak)

Komponenta	Naziv	Opis
Napajanje	USB-C priključak	9 V do 12 V DC putem USB-C s PD, do 30 W (minimalno 2 A)
	GPIO napajanje	5 V DC putem GPIO priključka (minimalno 3 A)
	PoE	Zahtijeva poseban PoE HAT
GPIO	40-pinski GPIO	GPIO, CAN, DMIC, I2C, I2S, PWM, SPI, UART itd.
	Boot način	QSPI Nor Flash, SDIO3.0, eMMC, UART
Ostalo	Gumb za resetiranje	Držanjem dulje od 3 s resetira sustav
	Dimenzije	100×74 mm
	Sukladnost	RoHS, FCC, CE
	Temperatura	0-50
	Debug funkcija	UART TX i UART RX preko 40-pinskog GPIO

Za potrebe rada je naručena verzija s 8 gigabajta RAM memorije i dodatnim ventilatorom5.1. s hladnjakom s ciljem smanjenja radne temperature i povećanja performansa. Debian je korišten kao OS pošto za njega postoje već izgrađene inačice direktno od StarFive-a, a podignut je na SD kartici od 32 gigabajta. U svrhu spajanja na internet je korišten USB WiFi priključak 5.1. Za dodatnu pohranu i brže pisanje i hvatanje podataka je korišten SSD od 128 gigabajta. Integrirani GPU podržava Vulkan 1.3, OpenGL ES i OpenCL 3.0. Valja napomenuti kako GPU trenutačno nema mogućnosti bilo koje vrste vektorskog procesiranja (RVV) što narušava performanse SBC-a. Proces pokretanja SBC-a je detaljno obrađen u dokumentaciji koju StarFive nudi tako da neće biti opisan u sklopu ovog rada.



Slika 5.2. I/O priključci računala

5.2. Problemi sa SBC-em

Pošto je VisionFive 2 nov proizvod na tržištu kojemu softver nije u svojoj finalnoj verziji (nedostaju određeni driveri i većina programa nije optimizirano), očekivano je naići na probleme vezane uz performanse, kompatibilnost određenih programa i slično. U nastavku su opisani temeljni problemi koji su rezultirali znatnim usporavanjem napretka ovog rada i kako su razriješeni (ukoliko jesu).

5.2.1. Manjak dostupnih informacija

Iako je StarFive-ova dokumentacija opsežna i nude gotove primjere programa koji koriste algoritme prepoznavanja objekata i ispunjavaju ostale zadatke računalnog vida, navedeni programi koriste zastarjele Python pakete koji već dolaze izgrađeni za VisionFive 2 ali onemogućuju testiranje najnovijih algoritama. Tako postoji OpenCV Python paket verzije 4.6.0. koja ne pruža mogućnost korištenja novih modela za prepoznavanje objekata kao što je YOLO v8. Za korištenje novih modela potrebno je izgraditi paket iz izvor-

nog koda pomoću cmake-a što je proces koji traje okvirno 6h. Izgradnja OpenCV-a je bila ponovljena nekoliko puta krivo postavljenih ili nedostajućih zastavica što je rezultiralo lošim performansama algoritama (inferencija od 30 sekundi na YOLOv5 modelu na kojemu je ranije postizana inferencija od cca 1 sekunde) ili rušenja samog procesa. Problem se ponovio prilikom izgradnje TensorFlow Lite paketa koji uopće ne postoji predizgrađen za RISC-V. Onnxruntime paket se ispostavio nemogućim za instalirati zbog konflikata sa sustavom. Povećanjem korisnika RISC-V sustava i online društva će s vremenom problemi ove prirode nestati pa će i korištenje VisionFive-a i sličnih SBC-a postati znatno jednostavnije i prijateljski nastrojeno prema novim korisnicima i osobama koje tek ulaze u svijet ugrađenih sustava.

5.2.2. Nasumično rušenje sustava i nemogućnost podizanja SBCa

SBC bi se povremeno znao srušiti i ugasiti sam od sebe, nakon čega bi bilo potrebno pokušati ga upaliti više puta prije nego li bi se sustav ponovno podigao. Ovo ponašanje je bilo nasumično i neovisno o tome je li sustav pod opterećenjem ili je samo upaljen i miruje. Ovo ponašanje je bilo dovoljno rijetko da nije uzrokovalo smetnje u samom razvoju projekta tako da direktni uzrok i popravak rješenja nikada nisu traženi. Pošto je računalo imalo dodatan ventilator i hladnjak za procesor, valja pretpostaviti da se ne radi o problemu pregrijavanja već da su u pitanju problemi s napajanjem (potencijalan pad struje) ili da problem leži u samom računalu i njegovom hardveru ili softveru.

5.2.3. GPU driveri

Imagination Technologies je firma koja je proizvela IMG BXE-4-32 MC1 GPU koji se nalazi u VisionFive-u, a samim time su zaduženi i za razvoj driver programa za čip koji su dosada bili ažurirani i nadograđeni svakih nekoliko mjeseci. Usprkos tome, postoje razni problemi s performansama čak i kod bazičnih zadataka poput reprodukcije 4K videa (za što je čip na papiru sposoban) čije učitavanje može potrajati i nekoliko minuta. Iako za GPU tehnički podržava Vulkan, OpenGL ES i OpenCL API, od navedenih trenutačno je u svrhu strojnog učenja i računalnog vida moguće koristiti isključivo OpenCL.

5.2.4. Nedostupnost softvera

Kako je ranije napomenuto, RISC-V je podosta nov u svijetu procesorskih arhitektura i kao takav nije široko prihvaćen, pa samim time ni firme ni individue nemaju velik interes, a ni mogućnost razvoju i prilagodbe softvera arhitekturi. Ova činjenica je uzrokovala velik broj prepreka u projektu pošto većinu modernih softverskih biblioteka (poput ranije spomenutog OpenCV-a i LiteRT-a) izgraditi pomoću CMake-a. Ovaj proces oduzima puno vremena.

5.2.5. Nemogućnost korištenja virtualnih okruženja uključujući pipx

Korištenje softvera "pip" za Python se pokazalo nemogućim pošto je Python instaliran na razini cijelog sustava. Zastavica "–break-system-packages" bi rezultirala errorom. Osim korištenja apt-get naredbe za instalaciju paketa, preostala je mogućnost postavljanja virtualnog okruženja pomoću naredbe venv ili korištenjem pipx paketa koji provodi postavljanje virtualnog okruženja u pozadini i zatim preuzima i instalira paket. Obje varijante su se pokazale beskorisnima jer bi se nakon pokretanja vrtjele po pola sata nakon čega bi se podigao error.

5.3. LiteRT Benchmark

U svrhu testiranja mogućnosti VisionFive-a i procjene performansi popularnih modela za prepoznavanje objekata, korišten je Googleov LiteRT (prijašnje TensorFlow Lite) paket koji je namijenjen mobilnim i rubnim uređajima, a primarna namjena mu je treniranje, testiranje, optimizacija i izvršavanje modela strojnog učenja. LiteRT također podržava hardversko ubrzanje korištenjem GPU-a. OpenCL je također podržan te je inicijalna ideja ovog testiranja bila vidjeti kolike su razlike pri korištenju CPU i GPU delegata na najpopularnijim modelima za računalni vid i zatim implementirati jedan od modela u pravom vremenu pomoću web kamere s GPU ubrzanjem. Pošto ne postoji predizgrađen LiteRT paket dostupan putem apt-get okruženja, paket je zajedno s dostupnim benchmarkom izgrađen pomoću CMake-a. Benchmark dolazi s predizgrađenim delegatima za OpenCL i pisan je u C++ jeziku dok je za svrhu rada korišten Python. Za Python je potrebna .so datoteka koja sadrži GPU delegat, a gradi se isključivo pomoću Bazela.

Instalacija Bazela se ispostavila nemogućom pošto još ne postoji potpora za RISC-V. Radi ovoga, implementacija programa s prepoznavanjem objekata u pravom vremenu i njihovom klasifikacijom uz korištenje GPU ubrzanja nije bila moguća već se morao koristiti CPU delegat.

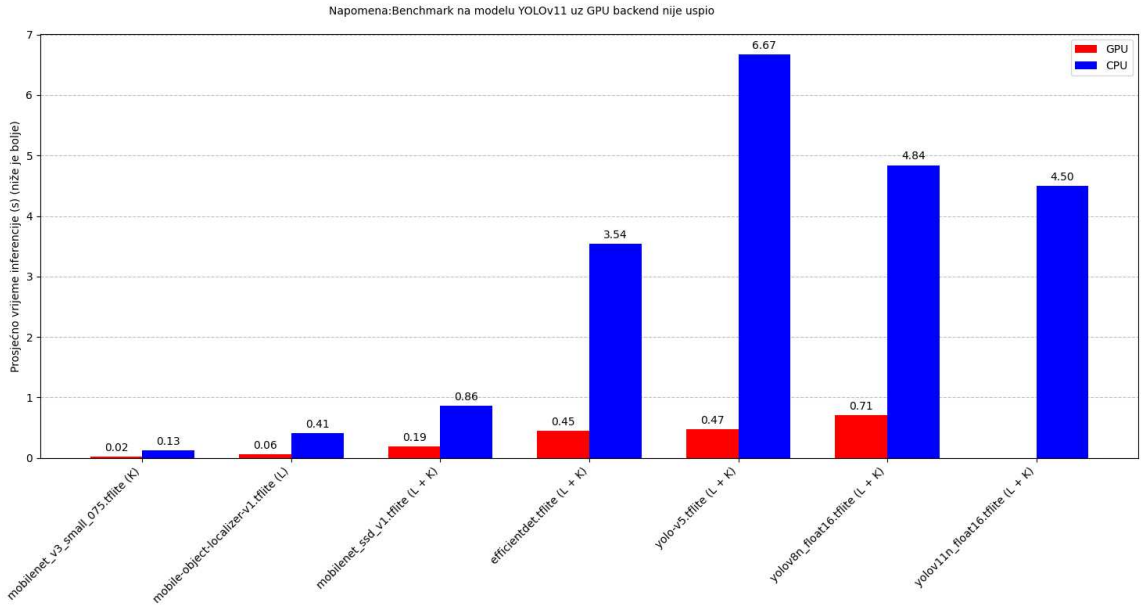
Tablica 5.2. Performanse različitih TFLite modela s i bez GPU delegata

Model	Funkcije	Delegat	Iteracije	Vrijeme inferencije (s)			
				Min.	Max.	Prosj.	Std. dev.
efficientdet.tflite	L + K	GPU	50	0.44	0.46	0.45	0.01
efficientdet.tflite	L + K	CPU	43	3.38	3.68	3.54	0.07
mobilenet_v3_small_075.tflite	K	GPU	50	0.02	0.03	0.02	0.00
mobilenet_v3_small_075.tflite	K	CPU	50	0.13	0.15	0.13	0.00
mobile-object-localizer-v1.tflite	L	GPU	50	0.06	0.07	0.06	0.00
mobile-object-localizer-v1.tflite	L	CPU	50	0.37	0.53	0.41	0.04
yolo11n_float16.tflite	L + K	CPU	34	4.44	4.57	4.50	0.03
yolo-v5.tflite	L + K	GPU	50	0.45	0.49	0.47	0.01
yolo-v5.tflite	L + K	CPU	23	6.56	6.78	6.67	0.05
yolov8n_float16.tflite	L + K	GPU	50	0.69	0.72	0.71	0.01
yolov8n_float16.tflite	L + K	CPU	31	4.80	4.89	4.84	0.03
mobilenet_ssd_v1.tflite	L + K	GPU	50	0.19	0.19	0.19	0.00
mobilenet_ssd_v1.tflite	L + K	CPU	50	0.68	0.94	0.86	0.08

Kolona "Funkcije" naznačava koje zadatke izvršava svaki od testiranih modela.

- **L** – Oznaka **L** predstavlja **lokalizaciju**. Model određuje poziciju objekata unutar slike i daje povratnu informaciju o njihovoj poziciji i granicama.
- **K** – Oznaka **K** predstavlja **klasifikaciju**. Model klasificira prepoznati objekt u jednu od klasa iz seta za trening.
- **L + K** – Oznaka **L + K** označava da model obavlja obje funkcije: **lokalizaciju** i **klasifikaciju**. Takav model može prvo odrediti poziciju objekta (lokalizacija) te

ga zatim klasificirati u odgovarajuću kategoriju.



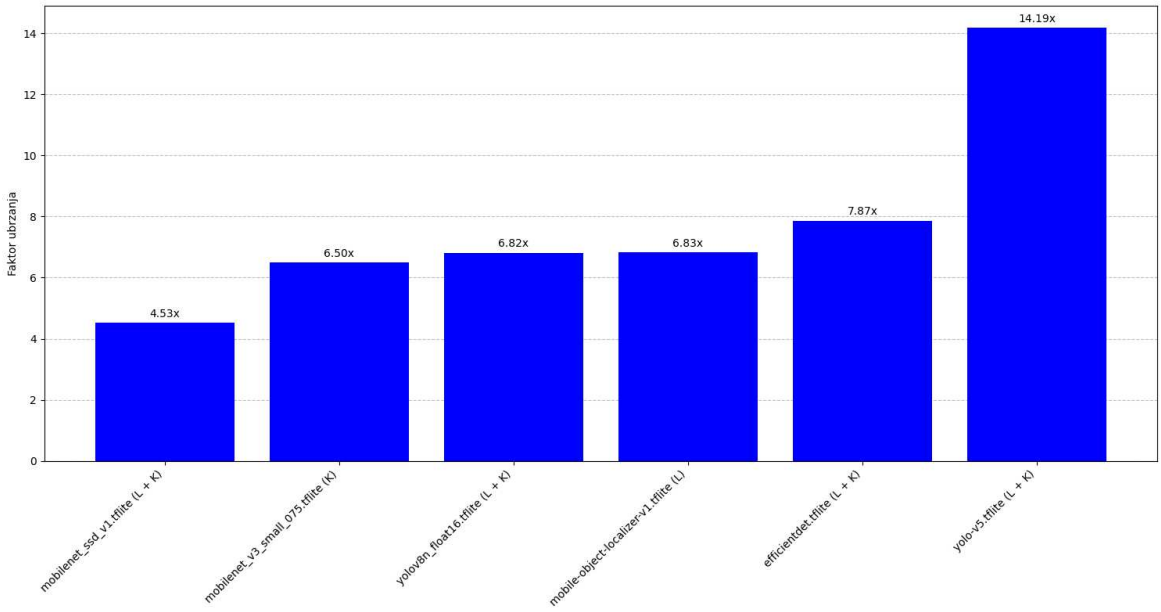
Slika 5.3. Usporedba prosječnog vremena inferencije GPUa i CPUa

Modeli koji izvršavaju samo jedan zadatak—bilo lokalizaciju ili kategorizaciju—postižu najkraća vremena inferencije, što je i očekivano s obzirom na njihovu jednostavniju neuronsku arhitekturu u odnosu na modele koji obavljaju oba zadatka istovremeno. U ovom slučaju, to su modeli **MobileNet V3** i **Mobile Object Localizer**, oba temeljena na MobileNet mreži.

MobileNet V3 predstavlja najnoviju verziju modela iz Googleove obitelji MobileNet i optimiziran je za klasifikaciju objekata, dok je **Mobile Object Localizer** prilagođen za lokalizaciju. Prema eksperimentalnim rezultatima, oba modela postižu vrijeme inferencije od svega nekoliko stotinki sekunde, što ukazuje na njihov visok potencijal za primjenu u scenarijima gdje je nužna obrada u stvarnom vremenu—pod uvjetom da je potreban samo jedan od dva zadatka.

S druge strane, složeniji modeli, poput **EfficientDet**, **YOLO v5**, **YOLO v8**, **YOLO v11n** i **MobileNet SSD V1**, koji kombiniraju lokalizaciju i klasifikaciju, pokazuju znatno duža vremena inferencije, osobito pri izvođenju na CPU-u. Među njima, pojedini modeli su optimizirani za rubne uređaje i mobilne platforme, što objašnjava njihovu relativno bolju izvedbu. Konkretno, **MobileNet SSD V1** je dizajniran za učinkovitu detekciju na uređajima s ograničenim resursima kao što je i VisionFive 2, čime postiže znatno bolje

rezultate u odnosu na ostale modele iz ove kategorije.



Slika 5.4. Faktor smanjenja vremena inferencije prilikom korištenja GPU-a

Kao što rezultati benchmarka pokazuju, korištenje GPU delegata je u prosjeku dovelo do približno 6.5× ubrzanja u odnosu na CPU delegate. U slučaju **YOLOv5** modela, ubrzanje je bilo još izraženije i iznosilo je približno 14×. Usprkos tome, prosječno vrijeme inferencije na GPU-u bilo je oko 0.5 sekundi, što je i dalje previsoko za potrebe detekcije objekata u stvarnom vremenu.

Usporedba vremena inferencije pokazuje da je **MobileNet SSD V1** pri izvođenju na GPU-u ostvario prosječno vrijeme inferencije od 0.19 sekundi, dok je **EfficientDet** na GPU-u postigao prosječno vrijeme inferencije od 0.45 sekundi. To znači da je **MobileNet SSD V1** približno 2.37 puta brži od EfficientDet modela, što ga čini očitim izborom za daljnja ispitivanja.

Zbog toga će se u nastavku rada detaljnije analizirati jedino **MobileNet SSD V1**, dok se modeli iz **YOLO** obitelji neće uzimati u obzir jer su previše spori za izvođenje u stvarnom vremenu na dostupnom hardveru. Također, model **EfficientDet** neće biti dalje testiran budući da je **MobileNet SSD V1** ostvario više nego dvostruko bolje rezultate.

5.4. PVRTune

PVRTune je alat za profiliranje i analizu performansi i opterećenja računalnih komponenti. Razvila ga je kompanija Imagination Technologies, koja je također razvila PowerVR GPU korišten na računalu VisionFive 2, stoga je PVRTune bio logičan izbor za praćenje performansi sustava. Ovaj alat omogućuje real-time nadzor rada GPU-a, prikazujući ključne metrike poput utilizacije shadera, opterećenja memorijske sabirnice, potrošnje energije i općenitog radnog statusa GPU jedinice.

5.4.1. Metodologija korištenja PVRTune-a

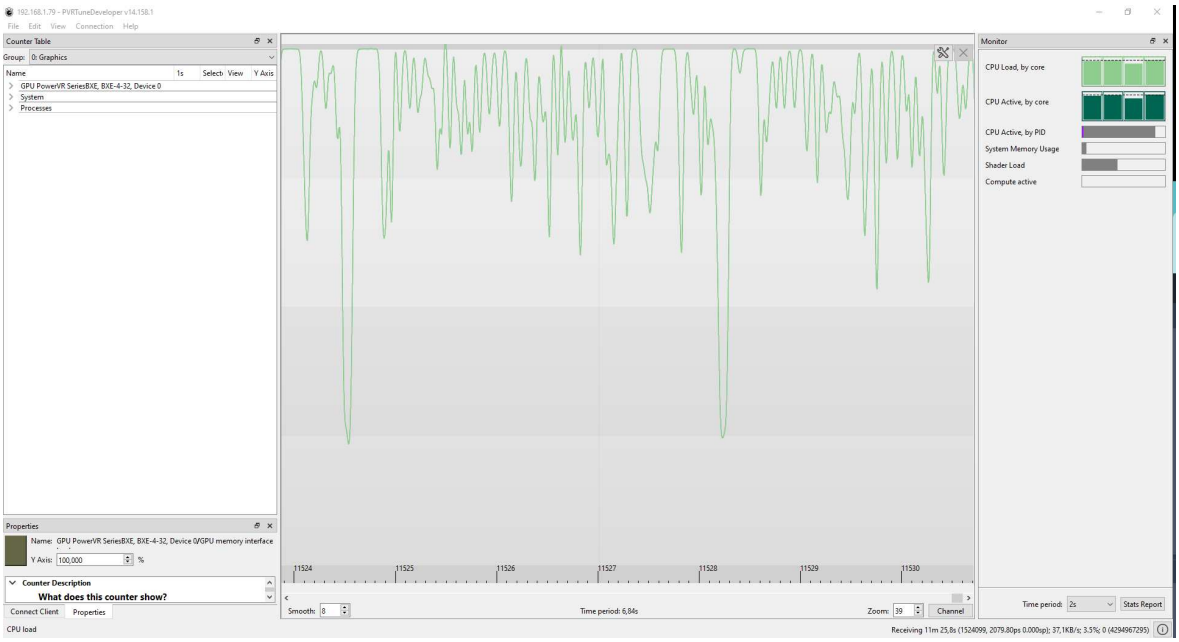
Primarna svrha korištenja PVRTune-a u ovom radu bila je provjera utilizacije GPU-a prilikom korištenja OpenCL backenda za izvršavanje zadataka računalnog vida. Budući da je PVRTune alat otvorenog koda, za njegovo korištenje bilo je potrebno samo kreirati korisnički račun i prijaviti se kako bi se dobio pristup svim potrebnim dokumentima i funkcionalnostima.

Za povezivanje s VisionFive 2 korišten je SSH protokol, koji je integriran unutar samog PVRTune-a. Međutim, prilikom prvog pokretanja alata, GPU driveri nisu bili prepoznati, što je onemogućilo nadzor opterećenja i performansi GPU-a. Ispostavilo se da VisionFive 2 "aktivira" drivere samo kada je uređaj spojen putem HDMI priključka ili kada je kamera povezana na računalo. Pri spajanju na HDMI, opterećenje GPU-a bilo je prepoznato isključivo kada bi se događale promjene na ekranu, dok ostali GPU zadaci nisu bili registrirani. S druge strane, kod spajanja kamere, svaka aktivnost GPU-a je bila registrirana i sve metrike su bile dostupne

5.4.2. Analiza opterećenja GPU-a i CPU-a

Za evaluaciju performansi sustava napravljena su dva eksperimenta:

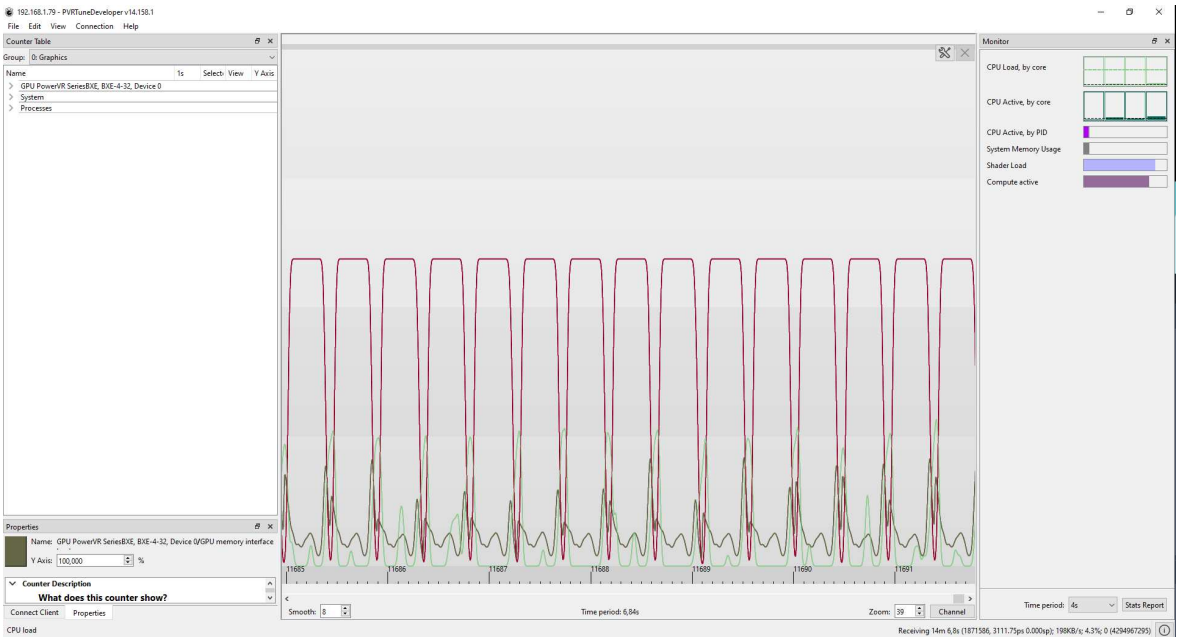
- Prvi eksperiment prikazuje opterećenje GPU-a i CPU-a kada OpenCL backend nije uključen.
- Drugi eksperiment prikazuje opterećenje GPU-a i CPU-a kada je OpenCL backend aktivan.



Slika 5.5. Opterećenje GPU-a i CPU-a bez OpenCL backenda

Na slici 5.5. prikazano je opterećenje GPU-a i CPU-a kada se zadaci izvršavaju isključivo na CPU-u, bez korištenja OpenCL backenda. Vidljivo je da GPU nema značajno opterećenje, što znači da se ne koristi za izvođenje računskih operacija.

Međutim, CPU pokazuje visoko i promjenjivo opterećenje, što ukazuje na to da su svi dostupni procesorski resursi angažirani za izvršavanje računskih zadataka. Budući da VisionFive 2 posjeduje četverojezgreni CPU s podrškom za osam procesorskih dretvi, može se potvrditi da se LiteRT benchmark pokretao na sve četiri jezgre.



Slika 5.6. Opterećenje GPU-a i CPU-a uz OpenCL backend

Na slici 5.6. prikazano je opterećenje GPU-a i CPU-a kada je OpenCL backend aktivan, što znači da se računanje sada delegira GPU-u umjesto da se izvršava isključivo na CPU-u.

Vidljivo je da GPU postiže gotovo 100% iskorištenja, pri čemu njegova operativna frekvencija doseže maksimalnih 600 MHz, što potvrđuje da se GPU koristi za izvođenje računskih operacija. Ovaj rezultat potvrđuje učinkovitost OpenCL backenda u prebacivanju tereta s CPU-a na GPU, čime se značajno rasterećuje CPU.

Istovremeno, CPU opterećenje je znatno smanjeno, što znači da CPU sada samo orkestrira zadatke i upravlja komunikacijom između memorije i GPU-a, dok sama računska obrada većinom ostaje na GPU-u. Ovakav način rada omogućuje bolju skalabilnost i efikasnije korištenje dostupnih resursa.

5.4.3. Zaključak

Korištenjem PVRTune-a omogućeno je precizno mjerenje performansi VisionFive 2 sustava, s naglaskom na optimizaciju GPU opterećenja. Pokazano je da:

- VisionFive 2 ne pokreće GPU drivere automatski, već zahtijeva HDMI priključak ili spojenu kameru.

- Bez OpenCL-a, CPU nosi cijeli teret obrade, dok GPU ostaje neaktivan.
- Uz OpenCL, GPU postiže punu iskorištenost, čime se poboljšava ukupna efikasnost sustava.

Ovo istraživanje potvrđuje važnost PVRTune alata u analiziranju i optimizaciji embedded GPU sustava poput onog na VisionFive 2, omogućujući bolje razumijevanje rada i potencijalne optimizacije algoritama računalnog vida.

5.5. Potencijalno ubrzanje pomoću AI čipa

Trenutačno na tržištu postoji nekoliko čipova čija je primarna svrha ubrzanje AI modela, a neki od njih su relevantni za zadatke prepoznavanja objekata opisane u ovom radu. Jedan od mogućih uređaja za ubrzanje ovih zadataka je Hailo 8 AI accelerator, koji nudi značajne prednosti u pogledu optimizacije izvršenja modela za duboko učenje, posebno u kontekstu rubnih uređaja.

Ovakav čip rješava sve ranije navedene probleme u vezi GPU-a VisionFive 2 i probleme s kompatibilnosti s popularnim paketima za računalni vid.

Hailo 8 nudi značajnu prednost u pogledu performansi. Prema dostupnim benchmark podacima, ovaj akcelerator može postići do 26 TOPS (trilijuna operacija po sekundi), što omogućava značajno ubrzanje inferencije u usporedbi s tradicionalnim CPU-ovima. Osim toga, Hailo 8 pokazuje visoku energetsku učinkovitost i nisku latenciju, što ga čini izuzetno pogodnim za rad u realnom vremenu na uređajima poput VisionFive 2. Integracija Hailo 8 s VisionFive 2 platformom moguća je putem M.2 portova, što omogućuje jednostavno povezivanje bez potrebe za dodatnim hardverskim modifikacijama.

Cijena Hailo 8 akceleratora na tržištu iznosi približno 200 EUR, što ga čini konkurentnim u odnosu na druge opcije na tržištu. S obzirom na visoke performanse koje nudi, ali i pristupačnu cijenu, Hailo 8 predstavlja vrlo isplativu opciju za implementaciju AI rješenja na rubnim uređajima.

6. Maksimizacija iskorištavanja hardverskih resursa

Jedan od temeljnih ciljeva ovoga rada bio je maksimalno iskoristiti hardverske resurse u svrhu ubrzanja izvođenja zadataka prepoznavanja objekata. Zbog mnogih ranije spomenutih prepreka, poput nedovršenih GPU drajvera, nemogućnosti izgradnje GPU delegata za LiteRT, kao i problema prilikom učitavanja neuronske mreže modela pomoću OpenCV-a (što nije uspjelo kod nijednog modela koji vrši i zadatke lokalizacije i kategorizacije osim YOLO, ali je inferencija na njima toliko spora da nije bilo svrhe istraživati dalje mogućnosti), a i manjka iskustva s novom platformom kao što je RISC-V i općenito rubnim uređajima, ovaj cilj ostaje neizvršen.

Prilikom traženja radova i internet blogova koji pokrivaju sličnu temu, nisam našao nikoga tko je u potpunosti uspio izvršiti ovaj zadatak, iako nekolicina radova poput [7] tvrde da jesu, iako su rezultati prilikom inferencije na CPU i GPU gotovo identični, što ukazuje na to da GPU nikada nije bio pravilno iskorišten. Također, valja napomenuti da je Starfive sam izdao dva koda koji služe kao primjeri za prepoznavanje objekata, ali niti jedan od njih ne koristi GPU.

Rad na VisionFive 2 računalu trenutačno iziskuje mnoge probleme, kako zbog manjka kompatibilnosti s drugim softverima, tako i zbog nedovršenog softvera za samo računalo. Ovakvi problemi bi s vremenom trebali postati znatno lakši jer će popularni paketi dobiti podršku za RISC-V procesore, a postojat će više dokumentacije online. Samim time, ovakav zadatak u budućnosti bi trebao biti trivijalan, dok se u trenutačnom razvojnom stadiju Starfive VisionFive 2 pokazao suprotnim.

7. Zaključak

VisionFive 2 pokazuje potencijal kao RISC-V platforma za primjenu u računalnom vidu, ali još uvijek zahtijeva značajna poboljšanja kako bi se mogao smatrati ozbiljnim konkurentom popularnijim SBC rješenjima poput Raspberry Pi-a. Ključan izazov trenutačno leži u nedovršenoj podršci za GPU akceleraciju, čime se ograničava iskorištavanje hardverskih kapaciteta kod zadataka računalnog vida. Prvi korak u poboljšanju performansi bio bi dovršetak GPU drivera od strane Imagination Technologies, čime bi se omogućilo učinkovitije inferenciranje neuronskih mreža poput MobileNet SSD i YOLO modela.

Osim toga, šira popularizacija RISC-V arhitekture neophodna je kako bi se povećala podrška za moderne biblioteke poput TensorFlow Lite i OpenCV. Trenutačno, veliki broj optimiziranih AI modela razvijen je prvenstveno za x86 i ARM arhitekture, dok RISC-V i dalje zaostaje u razvoju kompatibilnih softverskih rješenja. S vremenom, kako se ekosustav bude širio, očekuje se bolja integracija i podrška za računalni vid na RISC-V platformama.

Trenutačno stanje platforme sugerira da je VisionFive 2 primarno namijenjen iskusnim embedded developerima i entuzijastima koji su voljni ulagati dodatan trud u prilagodbu softvera. Za širu prihvaćenost unutar AI i računalnog vida, potrebno je ne samo poboljšati hardversku podršku već i pojednostaviti korisničko iskustvo, smanjujući prepreke koje trenutačno otežavaju razvoj i implementaciju modernih modela.

Unatoč ovim ograničenjima, prvi testovi s MobileNet SSD modelima pokazali su zadovoljavajuće rezultate, koji bi se dodatno mogli poboljšati s boljom podrškom za optimizaciju i akceleraciju modela. Nadalje, YOLO arhitektura, iako poznata po svojoj brzini i točnosti, trenutačno ne može biti u potpunosti iskorištena na VisionFive 2 zbog ograničenja u GPU podršci. S daljnjim razvojem drivera i povećanjem dostupnosti RISC-V

optimiziranih modela, ova platforma mogla bi postati konkurentan izbor za računalni vid u budućnosti.

Literatura

[1] Wikipedia contributors, “Scale-invariant feature transform (sift)”, https://en.wikipedia.org/wiki/Scale-invariant_feature_transform, [mrežno; stranica posjećena: veljača 2025.].

[2] J. Redmon, S. Divvala, R. Girshick, i A. Farhadi, “You only look once: Unified, real-time object detection”, *arXiv*, sv. 1506, str. 02640, 2015., [mrežno; stranica posjećena: veljača 2025.].

[3] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, i A. C. Berg, “Ssd: Single shot multibox detector”, *arXiv*, sv. 1512, str. 02325, 2016., [mrežno; stranica posjećena: veljača 2025.].

[4] K. Asanović, “The risc-v instruction set manual”, *Google Scholar*, 2014., [mrežno; stranica posjećena: veljača 2025.].

[5] RISC-V Foundation, “Risc-v history”, https://riscv.org/about/#history, [mrežno; stranica posjećena: veljača 2025.].

[6] StarFive Technology, “Visionfive 2 technical documentation”, https://doc-en.rvspace.org/Doc_Center/visionfive_2.html, [mrežno; stranica posjećena: veljača 2025.].

[7] L. Z. Loh, “Performance evaluation of starfive visionfive v2 and comparison on intel up board and raspberry pi 4b”, diplomski ili magistarski rad, UTAR, 2024.

Sažetak

Prepoznavanje objekata na RISC-V arhitekturi računala

Borna Budimir-Bekan

Računalni vid i detekcija objekata postali su ključni segmenti modernih sustava umjetne inteligencije, pri čemu se sve više istražuju alternativne hardverske platforme koje omogućuju njihovu izvedbu. Ovaj rad analizira mogućnosti primjene RISC-V arhitekture u kontekstu računalnog vida, s posebnim naglaskom na izvedbu detekcije objekata na VisionFive 2 jednopločnom računalu.

U prvom dijelu rada predstavljene su temeljne značajke RISC-V arhitekture te usporedba s popularnijim ARM i x86 rješenjima. Također se istražuju tehničke specifikacije VisionFive 2 uređaja, njegova kompatibilnost s modernim softverskim bibliotekama te ograničenja koja proizlaze iz nedovršene GPU podrške.

Eksperimentalni dio rada obuhvaća implementaciju i testiranje YOLO i MobileNet SSD modela na VisionFive 2. Analizirane su performanse u pogledu brzine inferencije na CPU-u te potencijalni dobitci uz buduću GPU akceleraciju. Rezultati pokazuju da iako VisionFive 2 može izvršavati osnovne zadatke računalnog vida, postoje značajna ograničenja u odnosu na konkurentske SBC uređaje poput Raspberry Pi-a, prvenstveno zbog nedostatka optimiziranih drivera i podrške za hardversku akceleraciju.

Zaključni dio rada ističe ključne izazove i moguća poboljšanja za buduću implementaciju računalnog vida na RISC-V platformama. Iako se RISC-V pokazuje kao perspektivna alternativa zatvorenim arhitekturama, potrebno je daljnje usavršavanje softverskog ekosustava kako bi se postigla bolja integracija s modernim AI modelima.

Ključne riječi: RISC-V; računalni vid; OpenCL; ugradbeni sustavi; VisionFive 2; LiteRT; YOLO; MobileNet SSD

Abstract

Object recognition on the RISC-V architecture

Borna Budimir-Bekan

Computer vision and object detection have become essential components of modern artificial intelligence systems, prompting increased exploration of alternative hardware platforms for their implementation. This paper examines the feasibility of utilizing RISC-V architecture for computer vision applications, with a particular focus on object detection performance on the VisionFive 2 single-board computer.

The first part of the paper introduces the fundamental features of RISC-V architecture and compares it with more widely adopted ARM and x86 solutions. Additionally, the technical specifications of the VisionFive 2 device are analyzed, including its compatibility with modern software libraries and the limitations arising from incomplete GPU support.

The experimental section involves the implementation and benchmarking of YOLO and MobileNet SSD models on VisionFive 2. The study evaluates inference performance using the CPU and explores the potential benefits of future GPU acceleration. The results indicate that while VisionFive 2 can handle basic computer vision tasks, it faces significant limitations compared to competing SBC devices like the Raspberry Pi, primarily due to the lack of optimized drivers and hardware acceleration support.

The conclusion highlights key challenges and potential improvements for the future development of computer vision applications on RISC-V platforms. Although RISC-V presents a promising alternative to proprietary architectures, further enhancements in its software ecosystem are required to achieve better integration with modern AI models.

Keywords: RISC-V; computer vision; OpenCL; embedded systems; VisionFive 2; LiteRT; YOLO; MobileNet SSD