Augmenting a Miniature Humanoid Platform with a Low-cost Networked Computer Vision Framework

Dylan Wallace[1], Blake Hament[2], Jean Chagas Vaz[2], and Paul Oh[3]

***Abstract*— Miniature humanoids are becoming an increasingly common platform for humanoid robotics research and education. However, the prohibitively high cost of advanced platforms such as the ROBOTIS-OP2 drives many educators and small research institutions toward cheaper options such as the ROBOTIS-Mini. While these platforms have versatility in full-body motion, they often lack computational power and vision capabilities. This paper presents the augmentation of the ROBOTIS-Mini with a camera, local processor, and networked system for computer vision. This augmented platform is referred to as Mini-CV. The Mini-CV system provides an ultra low-cost solution for computer vision that reduces the need for high on-board computational power and provides an advanced framework for networked control. A study of the latency in the system is presented and compared to that of the ROBOTIS-OP2, a popular miniature humanoid that retails for more than 20x the price of our augmented system. The results demonstrate the viability of the Mini-CV as an ultra low-cost alternative to more expensive miniature humanoid platforms.**

***Index Terms*— Humanoids, Computer Vision, Network, Low-cost, Latency.**

Fig. 1: ROBOTIS-OP2 (Left) and Mini-CV (Right)

I. INTRODUCTION

The price point of popular research platforms for full-size humanoid robotics ranges from hundreds of thousands to several million US dollars [1]. At such a high cost, these research platforms are not accessible to the population at large. Miniature, rather than full-sized, humanoid robots offer a more cost-effective alternative. Miniature humanoids, such as NAO or ROBOTIS-OP2, can serve as testbeds for algorithms that will later be ported to full-sized humanoids or they can be used to develop applications specifically for miniature humanoids. Even though these miniature humanoids are priced much lower than full-sized humanoids, they are still priced out of reach for anyone with a budget of less than several thousand dollars. In order to make humanoid robotics platforms more widely accessible to interested students and researchers, these platforms must be available at lower price points without sacrificing important capabilities.

Several ultra low-cost miniature humanoids have emerged in recent years that offer high degrees of freedom (DoF), but no vision capabilities. For instance, RoboPhilo Junior is a miniature humanoid robot platform with a price tag of $300 developed by RoboBrothers[4]. It is equipped with 10 servo motors and an ATmega32-16PU controller. Moreover the user can expand the original RoboPhilo's mobility by adding extra DoF. UBTech's Alpha 1S Humanoid Robot, costs around $500 and has 16 high precision servos. In addition, this robot uses a STM32-F103RDT6 processor with standard 128 MB external memory and Windows/iOS/Android system compatibility[5]. Despite these attractive features, the RoboPhilo Junior, Alpha 1S, and most other low-cost miniature humanoids do not have computer vision (CV) capabilities. This important gap should be addressed in order to give more researchers and teachers access to CV-enabled robotic testbeds and accelerate the development of CV-based algorithms and controls.

The ROBOTIS-Mini is a miniature humanoid platform developed by ROBOTIS at the accessible price point of 500 USD. Similar to Alpha 1S, ROBOTIS-Mini has 16 DOF and a built-in microcontroller. ROBOTIS-Mini was selected over Alpha 1S because of its RAM of 1GB and ability to interface with Android 2.3.3 (Gingerbread or greater), an increasingly popular OS among robotics researchers[2]. While the ROBOTIS-Mini is already an attractive, low-cost platform for research and education, it lacks vision capability. This paper proposes an inexpensive method of augmenting the ROBOTIS-Mini with computer vision capabilities such that the platform can be used for a greatly increased range of applications. With less than $50 in components, the already

---

[1] Dylan Wallace is an undergraduate student with the Department of Mechanical Engineering, University of Nevada, Las Vegas (UNLV), Las Vegas, Nevada 89119, USA `wallad3@unlv.nevada.edu`

[2] Blake Hament and Jean C. Vaz are PhD students with the Department of Mechanical Engineering, University of Nevada, Las Vegas (UNLV), Las Vegas, Nevada 89119, USA `blakehament@gmail.com` `chagasva@unlv.nevada.edu`

[3] Paul Oh is with Faculty of Mechanical Engineering, University of Nevada, Las Vegas (UNLV), Las Vegas, Nevada 89119, USA `paul.oh@unlv.edu`

[4] Robobrothers-Robotics, 4749 Bennett Dr, Livermore, CA, USA

[5] Ubtech-Robotics, 600 Wilshire Boulevard, Los Angeles, CA, USA

low-cost ROBOTIS-Mini can be equipped with a camera and networked computer vision system. This paper demonstrates the performance of our ultra low-cost augmented system with grayscale, Aruco marker detection, and facial detection trials, and compares the performance of our augmented ROBOTIS-Mini with that of the ROBOTIS-OP2.

This paper is structured as follows: Section II will describe the related work, focusing on low-cost humanoid platforms and networked computer vision systems; Section III will describe the platforms used for augmentation and experimentation; Section IV will describe the method of vision augmentation; Section V will explain the experimental methods used to evaluate the augmented system; Section VI will showcase results of testing and evaluating the design; and Section VII concludes and will present future work for the project.

II. RELATED WORK

Implementations of popular computer vision algorithms on scaled-down humanoids like ROBOTIS-OP and Aldebaran NAO have already been demonstrated. For example, Bolotnikova *et al.* implemented a real-time face recognition system on the NAO humanoid with built-in 1.22 MP camera [3]. They demonstrate the Viola-Jones face detection framework and robust image processing. With block processing of local binary patterns combined with a facial database, the system was successful in real-world facial recognition experiments involving subjects stationed between 1/2 and 2 meters from the robot.

Figat *et al.* developed improved glyph detection for the NAO's vision system [4]. They demonstrated their improved algorithm with detection of QR-codes and Aldebaran's NAO-marks. The authors found that QR-codes produced more accurate distance-to-landmarks measurements than NAO-marks. They report measurements from QR-codes had average relative error below 3%, whereas NAO-marks can generate as high as 9% error.

Fenn, Mendes, and Budden optimized the nonfunctional requirements of a miniature humanoid CV system and found that the functional performance of the system was also improved [5]. They developed an improved CV software architecture and tested the system with both the Aldebaran NAO and a Raspberry Pi with mock sensors and actuators. Although they introduced increased overhead by using wrappers and controllers instead of a simple pipeline, the overall performance was improved by dynamic selection of nodes. By focusing on modifiability, extensibility, and portability, Fenn *et al.* developed an improved CV system for miniature humanoids that is hardware independent and easily implemented at low-cost.

These projects were strictly software augmentations. They made use of the NAO robot's two RGB cameras. However the two cameras do not overlap, and therefore cannot be used for traditional stereo vision applications. To address this, Nefti-Meziani *et al.* developed a binocular vision system for the NAO as a low-cost hardware and software augmentation [6].

In the spirit of Nefti-Meziani *et al.'s* work, this paper presents hardware and software augmentation of an already low-cost system such that more researchers, educators, and students will have access to humanoid robotics platforms with powerful CV capabilities. Existing CV implementations on miniature humanoids have produced strong results, but the platforms themselves, like the NAO and ROBOTIS-OP, are still priced at approximately $10,000 per unit. Our system can be reproduced at a price point of less than $1,000, an order of magnitude less. Such a drastic reduction in price will enable more research and educational organizations to purchase the equipment and contribute to the robotics community.

This paper uses latency as the primary metric for evaluating the performance of our augmented ROBOTIS-Mini, henceforth referred to as Mini-CV. The Mini-CV system processes image frames on a networked computer, and we build on prior work involving the development and analysis of networked systems. Of particular interest, Hill *et al.* developed a method for measuring the latency of IP surveillance cameras over a network [7]. They present a method of measuring total latencies of a variety of networked cameras, utilizing ntp synchronization to measure the network latencies. Most importantly, they identify and analyze all of the latency components present in a networked camera system and conclude that, while analog cameras do have lower latencies, IP cameras are well within the tolerance for real-time processing. In developing and testing this networked system, we use latency analysis methods similar to Hill *et al.'s* method for measuring the latency of networked video surveillance systems.

III. HARDWARE

Miniature humanoid studies are a trending topic in the robotics community. The appeal of miniature humanoids is their low cost and high resistance to damage due to normal tripping and toppling. Additionally, they are lightweight and very mobile. Thus, miniature humanoids are attractive as low-risk testbeds for the development of full-sized humanoid applications, as well as for development of miniature-specific applications. These miniature-specific applications might involve service tasks in which small size is not a hindrance or maintenance tasks in which small size is an advantage.

To test and evaluate the proposed system, we compare the performance of our augmented system with that of a significantly more expensive miniature humanoid from the same manufacturer. The ROBOTIS-OP2 is already equipped with CV capabilities and serves as the control.

A. ROBOTIS-OP2

ROBOTIS-OP2 (OP2) is the second in the OP series. Its predecessor, commonly referred to as DARwIn-OP, has been extensively cited in miniature humanoid research publications [8] [9] [10]. The OP2 is a high-cost bipedal miniature humanoid weighting 2.9 kg and it is 0.454 m tall [11]. In addition, this humanoid is equipped with 20 DoF, which allows for a variety of movements that can be performed by

TABLE I: ROBOTIS-OP2 vs ROBOTIS-Mini specifications.

Features	ROBOTIS OP2	ROBOTIS-Mini
Price	$10,000	$500
Built-in PC	Intel Atom N2600	OpenCM 9.04
Gyroscope	3-Axis	NONE
DoF	20	16
Accelerometer	3-Axis	NONE
Camera	2MP HD USB	NONE
OS	Linux Ubuntu 10.10	Embedded RTOS
Language	C++/Java	Embedded C (OpenCM)

the robot. OP2's high mobility and advanced computational power make for a compelling research and development platform with a $10,000 price tag. Table I compares these specifications with those of the much more modestly priced ROBOTIS-Mini. Priced at only $500, the ROBOTIS-Mini lacks some important capabilities relative to the OP2, such as gyroscope, accelerometer, and camera. However, this paper demonstrates the possibility of augmenting these capabilities with low-cost components. Institutions that may not be able to afford an OP2 or similarly-priced robot can augment ultra low-cost platforms like the ROBOTIS-Mini to achieve the same capabilities with a far smaller budget.

B. *ROBOTIS-Mini*

The ROBOTIS-Mini is a low-cost miniature humanoid robot developed by ROBOTIS, the makers of ROBOTIS-OP2 and other miniature humanoid robot platforms. The Mini has a significantly smaller form-factor than the OP2, measuring just 26 cm tall with an arm-span of 36 cm. The Mini was developed to support education on humanoid robotics, particularly for K-12 schools. The Mini can be controlled using a smartphone app, the R+ Motion software developed by ROBOTIS, or by directly programming the OpenCM9.04 microcontroller with an Arduino-style programming language. These features, along with its low price, make the Mini a great choice for K-12 educators that want to introduce students to humanoid robotics. However, the absence of any computer vision capabilities leaves a large gap in the educational and research value of the ROBOTIS-Mini.

IV. DESIGN APPROACH AND IMPLEMENTATION

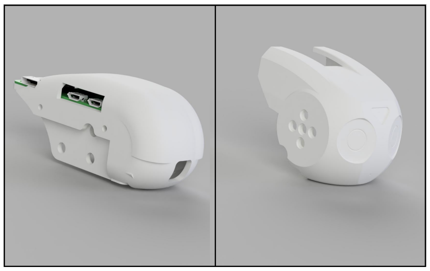


Fig. 2: New Mini-CV head (Left) and original ROBOTIS-Mini head (Right)

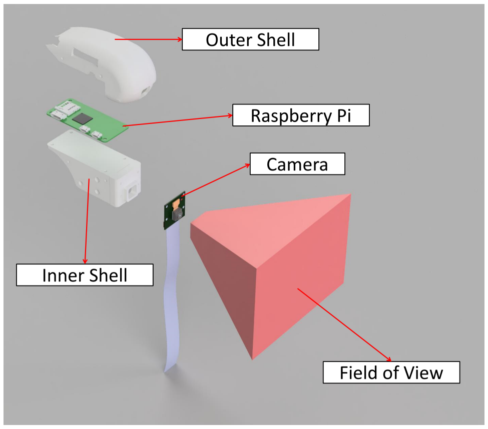


Fig. 3: Exploded view of Mini-CV head

In order to augment the vision capabilities of the ROBOTIS-Mini, a low-cost solution must be developed in order to keep the total cost of the augmented platform, nicknamed Mini-CV, as low as possible. To accomplish this, a networked approach for computer vision was adopted in this design. This allows the camera data to be acquired and encoded locally on the Mini-CV, and then streamed to a server for computer vision processing.

A popular solution for networked cameras is utilizing a Raspberry Pi camera to stream data to a local server or host on a web interface. For the Mini-CV, a Raspberry Pi Zero W was used with the Raspberry Pi camera to provide a low-cost, low-power solution for capturing visual data. The Raspberry Pi Zero W costs $10, the Raspberry Pi camera costs $20, and cabling for the system costs an additional $10 – bringing the total costs to $40 for the Mini-CV augmentation. To house these components, a lightweight head was designed and 3D printed, as seen in Fig. 2. The exploded view of the Mini-CV head design can be seen in Fig. 3, showing the inner and outer shells, and the electronic components for the 3D design.

The Mini-CV utilizes a networked approach to stream local camera data from the Raspberry Pi to a more powerful computer for vision processing. This takes a large computational load away from the Mini-CV, allowing more complex computer vision algorithms to be developed and used with the Mini-CV. To achieve the network stream, the ffmpeg package is used to capture individual frames from the Raspberry Pi camera and then host them on a local web page. This web interface also allows editing of the camera's settings in real-time. After each frame is sent to the web page, the processing computer captures the frame from the web page. This process is achieved by using the http protocol to grab the most recent frame from the web page. This essentially creates a live stream from the Mini-CV to the processing computer. However, due to the method of acquiring frames using http, the network delay is significantly reduced compared to traditional live-streaming

---

[6] trac.ffmpeg.org/wiki/StreamingGuide, ffmpeg streaming
[7] www.w3.org/Protocols/rfc2616/rfc2616.html, http protocol

methods. Once this stream is established on the processing computer, it can be used in the same manner as a local camera for vision processing. This allows for utilization of popular computer vision libraries such as OpenCV for processing with the Mini-CV. The architecture of the Mini-CV system is illustrated in Fig. 4.

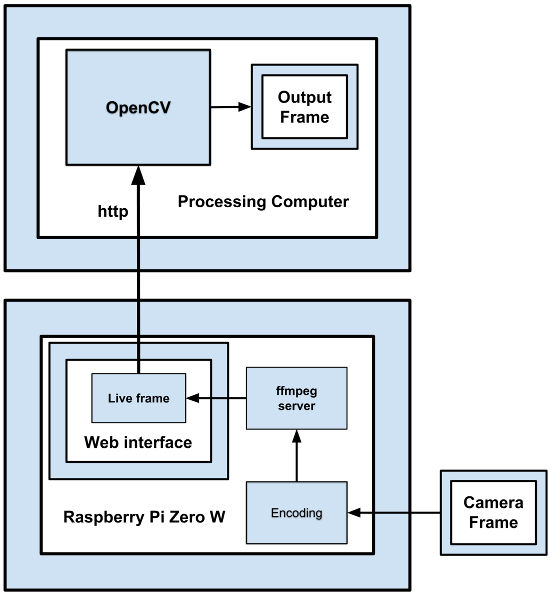


Fig. 4: Mini-CV architecture

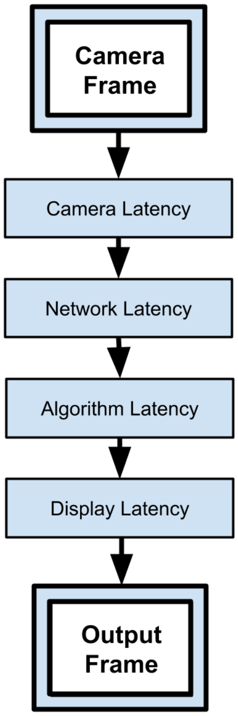


Fig. 5: Mini-CV latency pipeline

V. EXPERIMENTAL APPROACH AND SETUP

To evaluate the performance of Mini-CV, the latencies in the vision system are measured. There are many points in the network pipeline between receiving an image with the camera, and displaying the output of the vision processing. The full pipeline can be seen in Fig. 5. In order to test the full latency of the system, the latency within each computer vision algorithm also needs to be tested. This allows comparison of the true latency of the system with respect to other miniature humanoid computer vision systems, such as that of the ROBOTIS-OP2.

Due to the nature of the Mini-CV system's ffmpeg approach combined with the http transfer protocol, ntp synchronization is not the best-suited technique for measuring latency, and we must deviate slightly from the method of testing network latency described by Hill *et al.* [7]. To test the latency of the system, we use a method that involves the recording of an on screen timer. The timer, with accuracy to nearest millisecond, is displayed on the screen of the processing computer, along with the OpenCV output of the tested algorithm. The timer on-screen is recorded by the Mini-CV. While the timer and vision processing algorithm are running, a separate camera is used to record the two timer windows displayed on screen: one displaying the live timer, and one displaying the delayed timer. The difference between the delayed timer and the live timer gives the total latency of the pipeline, with mean value L_{Total}. Fig. 6 contains screen shots showing the simultaneous display of the live and processed on-screen timers.

The performance of the Mini-CV was compared to a non-networked approach for miniature humanoid computer vision, the ROBOTIS-OP2. The method for evaluating the OP2 was the same as the Mini, with the exception that the processing computer is onboard the OP2 itself, and the display is drawn directly from the OP2 using an HDMI connection. For experiments, both systems were situated so that the cameras' FOVs were centered upon the display, and the display was completely in view. The setup for experimentation can be seen in Fig. 7.

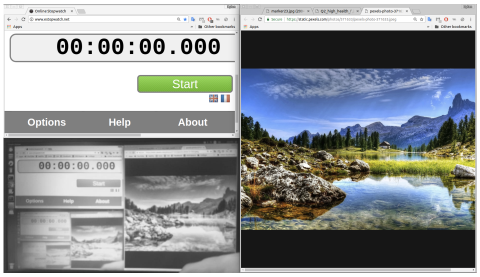
The hardware for the processing computer and wireless router for the Mini-CV are interchangeable. However, for this implementation an ASUS RT-AC1200 router, a consumer-grade high-speed router, was used. The processing computer used is an MSI GL62M 7REX-1067, with a dedicated Nvidia 1050Ti GPU. This hardware was chosen for its consumer, mid-range performance to get the most accurate representation of performance for the average user.

The Mini-CV was tested at three different input resolutions: 1920x1080, 1280x720, and 640x480. Due to software limiting of the OP2 to a maximum input resolution of 800x600, the OP2 was only tested at 640x480. For each resolution three computer vision algorithms were tested: AR

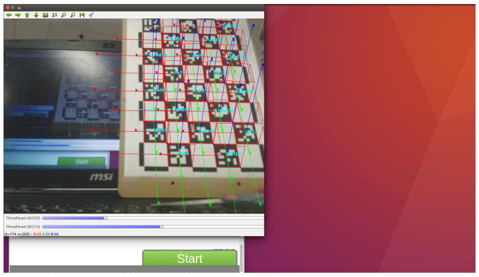
---

[8] opencv.org, OpenCV (Open Source Computer Vision Library)

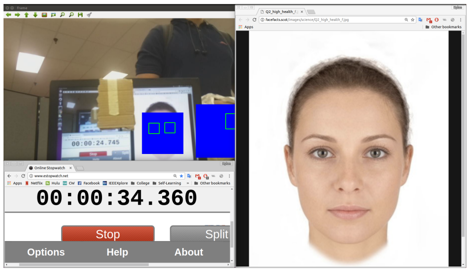
[9] www.uco.es/investiga/grupos/ava/node/26, ArUco Library



(a) Gray

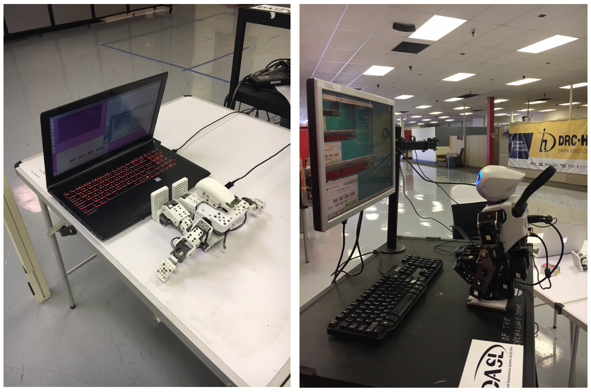


(b) Aruco



(c) Face

Fig. 6: Screen shot of experimental setup for Mini-CV



(a) Mini-CV

(b) ROBOTIS-OP2

Fig. 7: Physical layout of experimental setup for Mini-CV and ROBOTIS-OP2

marker detection, Viola-Jones face detection, and grayscale conversion as a baseline test. For each algorithm, four trials were performed, and all trials were averaged.

To determine the average network latency, $L_{Network}$ for the Mini-CV implementation, timing trials were performed with the grayscale algorithm. For each frame coming in, the time was recorded at the beginning of the processing loop, and again at the end of the processing loop for that frame. Taking the difference of these two times gives the algorithm latency, $L_{Algorithm}$. Four trials were performed, at 640x480, and the results were averaged to determine $L_{Algorithm}$ of the grayscale algorithm. This average value was subtracted from the L_{Total} of the grayscale trials at 640x480, along with the known camera latency (frame rate) L_{Camera} and display latency (refresh rate) $L_{Display}$ to determine the average network latency of the Mini-CV system. The equation below represents the average latency of the network, where all latencies are averages.

$$L_{Network} = L_{Total} - L_{Camera} - L_{Display} - L_{Algorithm} \quad (1)$$

VI. RESULTS

The results of the experiments described in Section V are shown for ROBOTIS-OP2 and Mini-CV in Tables II and III respectively. As these tables demonstrate, the Mini-CV significantly outperformed the ROBOTIS-OP2, with the exception of face detection. In the case of face detection, processing of each frame is more computationally expensive than a simple grayscale algorithm or Aruco marker detection. This first iteration of the Mini-CV included buffering of frames in order to minimize information loss. The ROBOTIS-OP2 does not buffer frames. Mini-CV's buffering was effective for the less computationally expensive algorithms, but it introduced significant lag with Viola-Jones face detection. To improve the performance of Mini-CV for face detection, the buffering could be reduced, or eliminated completely as is the case for ROBOTIS-OP2. The trade-off is frame loss, and the final design should be informed by the specific application in mind. This paper presents a general purpose platform that can be tailored accordingly.

TABLE II: ROBOTIS-OP2 trials.

Resolution	Algorithm	Run 1 [sec]	Run 2 [sec]	Run 3 [sec]	Run 4 [sec]
640x480	Aruco	1.6	1.5	1.5	1.5
	Gray	0.35	0.35	0.37	0.35
	Face	5.8	5.9	5.8	5.7

TABLE III: Mini-CV trials.

Resolution	Algorithm	Run 1 [ms]	Run 2 [ms]	Run 3 [ms]	Run 4 [ms]
1920x1080	Aruco	0.3	0.29	0.29	0.32
	Gray	0.34	0.29	0.28	0.34
	Face	12.71	12.5	4.42	11.5
1280x720	Aruco	0.21	0.25	0.23	0.21
	Gray	0.2	0.23	0.17	0.17
	Face	9.4	8.3	7.5	8
640x480	Aruco	0.25	0.25	0.23	0.25
	Gray	0.23	0.23	0.23	0.22
	Face	9.6	10.5	9.4	9.6

The improved performance of the Mini-CV system over ROBOTIS-OP2 can be attributed to the optimizations made in the Mini-CV's system with regard to capture, specifically the use of the ffmpeg method and the http transfer protocol. The ffmpeg method for encoding includes crowd-sourced optimizations for real-time encoding of video, as does the on-chip encoding for the Raspberry Pi camera. Furthermore, while one would expect a local transfer protocol such as the one used in OP2's system (PTP) to be faster than a network protocol, this is often untrue when unoptimized methods are used. The http capture method utilized in the Mini-CV system utilizes an efficient transfer of data from the Mini-CV's web server to the processing computer, ensuring lower network latencies than the OP2.

The results of the algorithm latency tests described in Section V are shown in Table IV. Using this data, the network latency of the Mini-CV can be determined according to Equation 1. The minimum, maximum, and mean values of each latency component are tabulated for the baseline test (grayscale, 640x480) in Tables V and VI for the Mini-CV and ROBOTIS-OP2 respectively.

TABLE IV: Grayscale algorithm latencies (640x480).

Run 1 ms	Run 2 ms	Run 3 ms	Run 4 ms	Average ms
49.26	33.86	35.59	35.48	38.55

TABLE V: Mini-CV Experimental latencies.

Latency Components	Min $[ms]$	Mean $[ms]$	Max $[ms]$
Camera	40	40	40
Display	17	17	17
Algorithm	34	39	49
Network	114	131	139
Total	220	227	230

TABLE VI: OP2 Experimental latencies.

Latency Components	Min $[ms]$	Mean $[ms]$	Max $[ms]$
Camera	40	40	40
Display	17	17	17
Algorithm	34	39	49
Network	244	259	279
Total	350	355	370

As seen in Tables V and VI, the latency of the network for the Mini-CV averages 131 ms, while the latency for the local network of the OP2 averages 259 ms. This shows that the Mini-CV has a significant performance increase over the ROBOTIS-OP2, with over 50% less latency in the Mini-CV system. These results show great promise for the Mini-CV system, as these latencies are very low for live-video streaming conditions and show significant improvement over the OP2 system.

VII. CONCLUSION AND FUTURE WORK

This paper presented the development and testing of an ultra low-cost computer vision augmentation of a miniature humanoid. The results of latency trials demonstrate the viability of the augmented system, Mini-CV, as an educational and research platform with computer vision capabilities. By networking Mini-CV with an off-board computer for image processing, the on-board processor is freed up for more vital tasks related to motion and balance. This paper demonstrates how the capabilities of low-cost miniature humanoids can be inexpensively augmented in order to make advanced robotics education and research more widely accessible.

Future work will involve the development of other hardware and software modules for Mini-CV that increase its appeal as a research platform. Future projects will involve: integrating full-body motion planning and complex walking algorithms; improving software architecture; adding sensors like gyroscope, accelerometer, and RGB-D camera; and implementing more advanced computer vision applications such as object recognition and Visual SLAM. Furthermore, with the ability to use off-board computers for vision-based processing, the authors hope to incorporate machine learning algorithms related to computer vision and humanoid robotics into Mini-CV's architecture.

REFERENCES

[1] ”Thirteen Advanced Humanoid Robots for Sale Today,” *Smashing Robotics,* April 16, 2016. [Online]. Available: https://www.smashingrobotics.com/thirteen-advanced-humanoid-robots-for-sale-today/. [Accessed: November 27, 2017].

[2] C. N. Thai, ”ROBOTIS-MINI System,” *In Exploring Robotics with ROBOTIS Systems,* pp. 157-172, 2015.

[3] A. Bolotnikova, H. Demirel and G. Anbarjafari, ”Real-time ensemble based face recognition system for NAO humanoids using local binary pattern,” *Analog Integrated Circuits and Signal Processing,* pp.467-475, 2017.

[4] J. Figat, and W. Kasprzak, ”NAO-mark vs QR-code Recognition by NAO Robot Vision,” *Automation, Robotics and Measuring Techniques,* pp. 55-64, 2015.

[5] S. Fenn, A. Mendes and D. M. Budden, ”Addressing the non-functional requirements of computer vision systems: a case study,” *In Machine Vision and Applications,* pp. 77-86, 2016.

[6] S. Nefti-Meziani, U. Manzoor, S. Davis and S. K. Pupala, ”3D perception from binocular vision for a low cost humanoid robot NAO,” *Robotics and autonomous systems,* pp.129-139, 2015.

[7] R. Hill, C. Madden, A.v.d Hegel, H. Detmold, and A. Dick, ”Measuring Latency for Video Surveillance Systems,” *In IEEE Conference on Digital Image Computing: Techniques and Applications,* pp. 8995, Melbourne, Australia, 2009.

[8] M. Gomez, B. Khaday, A. Gonzalez, A. Esmaeili and E. T. Matson, ”Golf-Playing DARwIn-OP: A Theoretical Approach,” *In Robot Intelligence Technology and Applications 4,* pp. 411-421, 2017.

[9] J. C. Vaz, H. Lee, Y. Jun and P. Oh, ”Towards tasking humanoids for lift-and-carry non-rigid material,” *In Ubiquitous Robots and Ambient Intelligence (URAI),* pp. 316-321, Jeju, South Korea, May, 2017.

[10] C. Teixeira, L. Costa and C. Santos, ”Biped locomotion-improvement and adaptation,” *In Autonomous Robot Systems and Competitions (ICARSC),* pp. 110-115, Espinho, Portugal, May, 2014.

[11] I. Ha, Y. Tamura, H. Asama, J. Han and D. W. Hong, *”Development of Open Humanoid Platform DARwIn-OP,”* In SICE Annual Conference, pp. 3190-3195, Tokyo, Japan, Sep, 2011.

[12] S. Nefti-Meziani, U. Manzoor, S. Davis and S. K. Pupala, ”Development of Open Humanoid Platform DARwIn-OP,” *In Robotics and autonomous systems,* pp. 129-139, 2015.

[13] L. George and A. Mazel, ”Humanoid robot indoor navigation based on 2D bar codes: Application to the NAO robot,” *In Humanoid Robots (Humanoids),* pp. 329-335, Atlanta, United States, October, 2013.