A Protocol Generator Tool for Automatic In-Vitro HPV Robotic Analysis

Juan Pedro Dominguez-Morales[1], Angel Jimenez-Fernandez[1], Saturnino Vicente-Diaz[1], Alejandro Linares-Barranco[1], Asuncion Olmo-Sevilla[2], and Antonio Fernandez-Enriquez[3]

[1] Robotic and Technology of Computers Lab., University of Seville, Av. Reina Mercedes s/n, 41012 Sevilla, Spain
{jpdominguez,alinares}@atc.us.es
[2] Technical Department, Master Diagnostica S.L., Granada, Spain
asuncion.olmo@vitro.bio
[3] Technical Department, VITRO S.A., Sevilla, Spain
antonio.fernandez@vitro.bio

Abstract. Human Papilloma Virus (HPV) could develop precancerous lesions and invasive cancer, as it is the main cause of nearly all cases of cervical cancer. There are many strains of HPV and current vaccines can only protect against some of them. This makes the detection and genotyping of HPV a research area of utmost importance. Several biomedical systems can detect HPV in DNA samples; however, most of them do not have a procedure as fast, automatic or precise as it is actually needed in this field. This manuscript presents a novel XML-based hierarchical protocol architecture for biomedical robots to describe each protocol step and execute it sequentially, along with a robust and automatic robotic system for HPV DNA detection capable of processing from 1 to 24 samples simultaneously in a fast (from 45 to 162 min), efficient (100% markers effectiveness) and precise (able to detect 36 different HPV genotypes) way. It includes an efficient artificial vision process as the last step of the diagnostic.

Keywords: Robotic arm · In-vitro analysis · Automatic diagnostic · Computer vision · OpenCV · Convolutional neural networks · XML protocol

1 Introduction

Human papillomavirus (HPV) is a group of more than 150 DNA viruses from the papillomavirus family. Most HPV infections will cause no physical symptoms, but others could cause benign papillomas or premalignant lesions that could lead to cancer, especially cervical cancer. HPV is transmitted through intimate skin-to-skin contact and is the most common sexually transmitted infection (STI). This virus is so common that nearly all sexually active men and women get it

at some point in their lives, making it a very high interest subject of study for cervical cancer screening.

Persistent human papillomavirus infection is a necessary cause for the development of cervical cancer [1,2]. Therefore, sensitive HPV detection is crucial to establish the risk of progression to high grade intraepithelial lesions and cervical carcinoma. In addition, HPV genotyping constitutes an important tool to assess the specific genotype-associated risk of progression and subsequent patient management.

HPV testing offers greater sensitivity for the early detection of pre-cancerous cervical lesions and has been recommended as a triage tool for efficient patient management. The HPV detection systems available on the market are mainly based on hybridization assays [3], signal amplification assays [4], or the amplification of viral sequences by Polymerase Chain Reaction (PCR) [4]. Several HPV detection systems do not identify the specific genotype(s) present in the sample, and their results have a limited usefulness as they can only be reported as high-risk positive, low-risk positive, or high-risk + low-risk positive. The market requires highly sensitive and specific HPV genotyping systems. These methods should also be automated, rapid and reliable.

There are several systems in the market for HPV DNA detection. Most of them are time consuming (6–8 h minimum to get results) (CLART® Technology from Genomica, Spain[1]), others are not fully automatic (FT-Pro Flow-Through hybridization system from Diagcor BioScience, China[2]), or are not able to identify a multiplex panel of HPV genotypes (Cobas from Roche Diagnostics, Switzerland[3]). Other systems are not able to run a set of samples simultaneously, but only one sample per run (GeneXpert from Cepheid, USA[4]).

HybriSpot 24 (HS24) platform[5] is a robotic device intended for automatic and rapid multiplexing analysis of biomarkers, by a reverse dot blot colorimetric hybridization into a porous CHIP based on DNA Flow Technology. This hybridization platform allows the target molecules (DNA amplicons) to vertically flow through and cross the membrane towards the complementary capture probes that are immobilized inside the matrix pores. This approach enables the reaction between the target molecules in a three-dimensional porous environment rather than passive surface hybridization, allowing the reactions to be completed in seconds. It generates much higher signal intensities in a very short period of time, reducing reagent volumes and processing times from hours to minutes, which represents an economic alternative for molecular diagnostic.

[1] Genomica, "CLART® Technology." [Online]. Available: http://genomica.es/en/in_vitro_diagnostics_products.cfm.

[2] Diagcor BioScience, FT-Pro Flow-Through hybridization system. [Online]. Available: http://www.diagcor.com/en/mdx-products/detail/ft-pro-auto-system.

[3] Roche Diagnostics, Cobas. [Online]. Available: http://molecular.roche.com/assays/Pages/cobasHPVTest.aspx.

[4] Cepheid, GeneXpert. [Online]. Available: http://www.cepheid.com/us/cepheid-solutions/systems/genexpert-systems/genexpert-i.

[5] *VITRO S.A., Hybrispot 24. [Online]. Available:* www.masterdiagnostica.com/products/moleculardiagnostickits/dnaflowtechnology/instruments.aspx.

The rest of the manuscript is structured as follows: Sect. 2 presents hybriSpot 24, which is the biomedical robot that is used in this work to implement and execute XML protocols, as well as to obtain execution times. Then, Sect. 3 explains the image analysis technique used in this work. Section 4 introduces XML protocols and describes their hierarchical structure and components (Basic Commands and High-Level Commands). Section 5 describes the process of building XML protocols. Section 6 presents results for a HPV diagnosis protocol in terms of execution times, performance, robustness and validation. Finally, Sect. 7 presents the conclusions of this work.

2 HybriSpot 24

HybriSpot 24 (See Fig. 1 is a fully automatic biomedical robot developed by VITRO S.A[6], DropSens[7], Master Diagnostica[8] and RTC Lab[9], and commercialized by VITRO S.A., whose main aim is to detect and genotype viruses like HPV. HS24 can process from 1 to 24 samples simultaneously in a period of time that ranges from 45 to 162 min, respectively. It provides a diagnostic platform for rapid detection and identification of HPV genotypes by reverse dot blot hybridization onto a microarray porous membrane (see Fig. 2). HS24 is based on a Gilson® GX-271 Liquid Handler robotic arm. This arm is a very rigid and robust Cartesian Coordinate Robot (CCR) that also has a dispenser, Fig. 1, which is a syringe-like electronic device that allows to pour or to suction liquids from various inputs automatically using specific commands. The dispenser is connected to a probe located on the robot, in addition to a washing solution container that is used to purge the system after particular steps of a protocol. HS24 also has:

- A washing station with two positions: a waste position and a probe cleaning position.
- Two 3×8 (three rows and eight columns) matrices to place DNA samples.
- 28 containers for reagents. Four of them are dedicated to a special reagent (hybridization buffer), which is most widely used in every protocol, and have two heaters so that the temperature can be set inside the containers. Setting this reagent to a specific temperature at a particular step of the protocol is essential for the speed and success of the analysis.
- Two 4×3-matrix (four rows and three columns) reaction chambers with one associated Peltier cell. Each chamber allows the incubation process on the membranes.

Using the robotic arm, the dispenser and the diverse elements presented above, HS24 is able to combine reagents with DNA samples, incubate them and execute a set of operations that will result on the spots color development at

[6] VITRO S.A. Available: http://www.vitro.bio/.
[7] DropSens. Available: http://www.dropsens.com/.
[8] Master Diagnostica. Available: http://www.masterdiagnostica.com/.
[9] RTC, Robotic and Tech. of Computers Lab. http://www.rtc.us.es/.

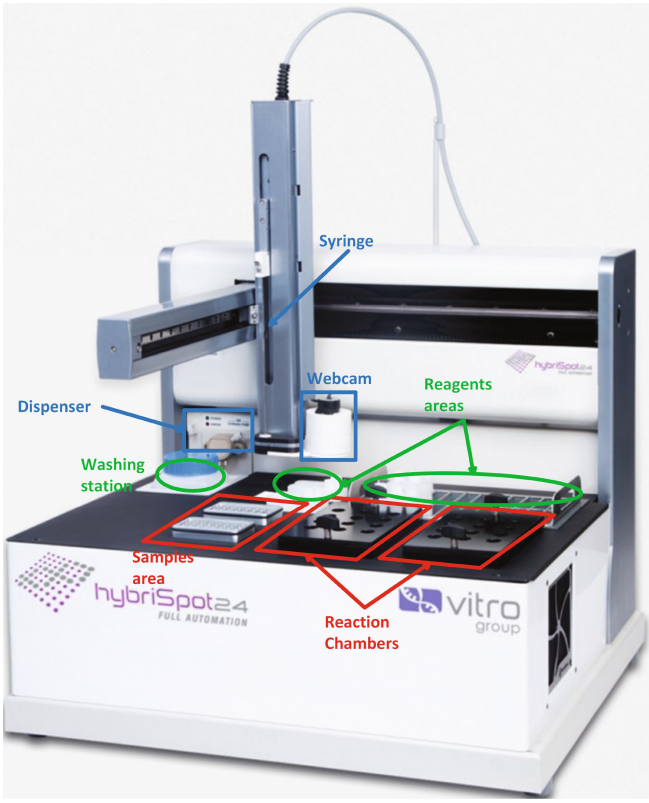

Fig. 1. HS24 instrument, composed of a Gilson® GX-271 Liquid Handler robotic arm with one dispenser attached, a washing station, a webcam, two reaction chambers with peltier cells, two reagents zones (with heaters), two 12 samples areas and embedded STM32-based USB PCBs for heaters / valves control.

the end of the process. In order to provide to the system with enough flexibility for other protocol implementations, or for tuning or debugging a protocol under development, all these operations are described in XML. Thus, the protocol is a sequence of XML sentences, which include information about every single execution step of the robot. Each membrane has a printed matrix with spots. Some spots are always there for control, some others will show up depending on the result of the detected genotypes. Figure 2 shows a membrane with highlighted spots after successfully completing a HPV diagnosis protocol. HS24 includes a high definition camera attached to the robotic arm that takes pictures of the membranes. They are sent to the host computer software, where they are automatically analyzed with two different mechanisms to perform a diagnosis and store it in a database: (a) OpenCV [5] artificial vision algorithms and (b) Convolutional Neural Network (CNN) previously trained with an extensive database collected in Master Diagnostica during the instrument testing period.

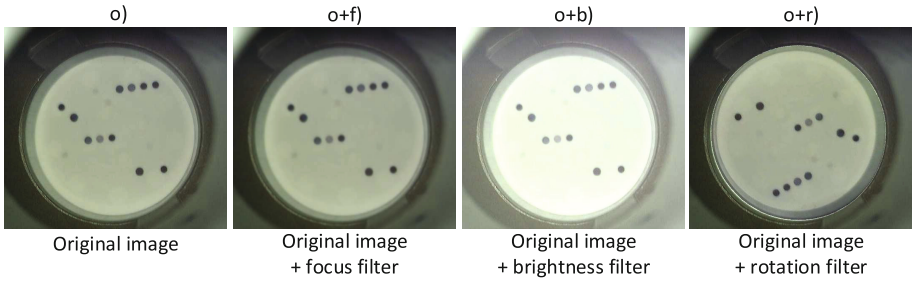

Fig. 2. Membrane with highlighted spots after executing a HPV diagnosis protocol (o). o+f, o+b and o+r are the output from different synthetic samples generation techniques.

3 Image Analysis

The principal approach implemented in this instrument is a classical one using artificial vision techniques, through the OpenCV library; we aim to compare this principal one to a deep learning based technique, using Convolutional Neural Networks. The second one aim is for validating deep learning in this particular application. The OpenCV library technique will be used as the ground truth for the deep learning training procedure, since 100% of accuracy has been obtained in the laboratory with proper camera configuration and light conditions. Once the protocol procedure of the samples have concluded, the camera installed in the instrument, close to the syringe, is activated and a picture is taken. The following operations are applied to the picture: (a) a smooth filter process, (b) a circle detection for the membrane plus centering it, (c) remove from the image anything not useful, (d) search for key spots in the membrane using template matching techniques for three different small circles detection, (e) distance calculation between these spots (the maximum distance identifies two of the five control spots, and the other three follow logic rules to be located), (f) membrane rotation to the same view for all membranes in the database, (g) diagnostic depending on how many spots and which ones are detected in the membrane. In the other hand, for the CNN technique, a database has to be created using the analyzed samples during the testing period of the instrument for the HPV analysis only. The images taken from the samples after the HPV analysis have been processed using different techniques like focus filters (o+f), brightness filters (o+b) and rotation filters (o+r) to increase the number of images in the database in a synthetic way. A classic CNN architecture extracted from LeNet-5 has been implemented and trained with Caffe [13]. The CNN execution can be performed in the HSHS software by using the open-source Caffe libraries.

4 XML Protocols

Our instruments have to execute a particular protocol to analyze a tissue sample to make a diagnostic of the presence of a particular DNA sequence

(i.e. HPV, Zoonosis, Sepsis, etc.). This protocol is stored as a sequential list of commands that the robot has to execute to perform a specific functionality. XML (eXtensible Markup Language) [6] has been selected in this case. XML is a language developed by W3C that is mainly used to store information. The main benefits of this language are its easy managing and its very legible structure. The information in this format is saved under a hierarchical structure similar to a tree: there is a root node and subtrees of children with a parent node. Thus, we call it XML-protocol. The root node is the name of the protocol, and two different types of children could be below the root (by convention, trees are drawn growing downwards):

- If the node is a leaf node (it does not have any children) then it is a Basic Command, i.e. a simplest actuation of the robot.
- If the node is an inner node (it has at least one child) then it is a High-Level Command. This command is a set of Basic Commands that describes a higher level functionality.

4.1 Basic Commands

They are instructions that describe basic functionalities of the robot, like position control, dispenser control, valves and pumps control and Peltier cells control. The set of basic commands that an XML-protocol can use allows to execute all the steps needed by a biomedical robot in the analysis process. These commands are:

- **RobotCmd**. Used to command the arm motion in the X, Y and Z Cartesian axes. The Z movement allow the syringe of the robot to go up and down in order to suction or dispense liquids properly. The robot has a liquid detector in the syringe. There are four additional parameters in this command that allow to (1) select movements only in X and Y axes, (2) do a homing process after the command is executed, (3) do a movement only in Z axis and (4) enable liquid detection in the probe. XY movements are allowed only if Z axis is under a security threshold to avoid syringe damage. As can be seen, the speed of the movement cannot be set with this command. The robot speed is controlled with a different parameter and cannot be modified using these kind of commands.
- **DispCmd**. Its purpose is to suction or dispense liquids like reagents or DNA samples using the probe attached to a dispenser. Quantities used in this command should be checked carefully to avoid exceeding the maximum capacity of the dispenser. The speed to execute this process can also be specified with a value between 0.01 and 40 ml/min, in addition to the position of the valve of the dispenser, which will allow to suction or dispense from the probe tip or from the washing solution container. The speed in the absorbing process should not be high, since the potential formation of air bubbles could affect the entire analysis [7].
- **PeltierCmd**. This command controls the temperature of the Peltier cells located bellow the reaction chambers and the main reagents of the robot.

This process is essential for the incubation of the samples. Some parameters are required for executing this command, such as the ID of the Peltier cell that will be affected, the target temperature, and the waiting time after reaching the temperature. This command will block the following command until this wait is over.
- **ValveCmd**. On a biomedical robot, valves redirect the liquid flow in the system. ValveCmd command allows this task, adding some control over it by using three parameters: (1) to specify the action (on/off) that will be applied to (2) a specific valve or to a set of them and (3) how much time will the system wait after this action until the next command starts its execution.
- **PumpCmd**. Pumps allow to turn on or off the liquid flow in the robot. This is very useful in washing and waste removal processes. This command has the same parameters as ValveCmd, but it acts on the pumps instead of the valves.
- **WaitCmd**. This command waits for a specific amount of time (minutes or seconds.
- **CameraCmd**. This command takes a picture of the membrane whose position is specified as an input. The captured picture is processed using an OpenCV (Open Source Computer Vision) [5] algorithm, or a CNN previously trained. OpenCV is an open source computer vision and machine learning software library of programming functions mainly used for motion tracking [8], facial or gesture recognition [9,10] and augmented reality [11]. This library has been used to detect which spots of the matrix that is printed on the membrane are highlighted, and their intensity. With this information, OpenCV gives an automated diagnosis of the detected genotypes.

As they are focused on low-level functionalities, creating an entire protocol using only these commands could be a slow and tedious process with a high rate of human error on it, which may cause serious damage to the robot.

4.2 High-Level Commands

High-level commands are a set of basic commands that describe their whole basic functionalities under a more complex and useful task. These commands allow the creation of longer protocols faster, and since they need less parameters to be specified, human mistakes in this process are reduced. The most important high-level commands are:

- **Wash**. This command cleans the probe attached to the robot both internally and externally, using a washing solution, which is pumped through the system using basic commands to control valves and pumps. Throughout the execution of the protocol, the robot uses different reagents and samples, which could spoil the whole process if they mix at some point. That is what makes a purge command like this so important for a biomedical robot.
- **Reagents dispensing**. Before a protocol starts, the camera attached to the robotic arm scans the workspace, to identify where the reagents needed in the

protocol are placed, and saves this information. Using this information plus the localization of the membranes that will be used (set by the user), this command is able to dispense a specific quantity of any of the reagents (both values are specified as parameters for this functionality) in every membrane used in the protocol. Figure 3 shows the execution flowchart for this command.

- **Samples dispensing**. This high-level command is similar to the reagents dispensing command described above. Instead of dispensing a specific quantity of a reagent in the membranes, it dispenses the reagent along with a small quantity of a sample, both specified as parameters.
- **Camera Capture**. After the protocol is finished, this command is launched to take images of the membranes that were used, and to analyze them with OpenCV algorithm. This command is a combination of multiple CameraCmd basic commands.
- **Incubation**. As the incubation of the membranes is one of the main steps in a protocol, this command was created for this purpose. The incubation high-level command combines a PeltierCmd and a WaitCmd. This will set the membranes to a temperature that is specified in the input parameters and then it waits for a specific amount of time after the Peltier cell reaches the target temperature.Incubation.
- **Drain**. After some particular steps of the protocol, the membranes need to be washed from reagents and samples that were dispensed. This cleaning process is done with the drain command, which uses pump and valve basic commands to allow the flow of washing solution into the system below the reaction chambers where the membranes are placed.
- **Remove waste**. Usually, the quantity of reagent that is suctioned is higher than the one that is dispensed. This adds a small security gap of reagent to maintain the pressure in the pipe that connects the probe with the dis-

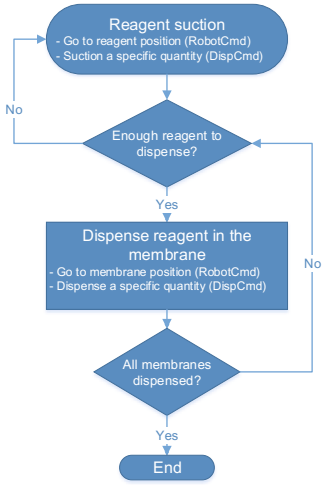

Fig. 3. Dispense Reagents command flowchart.

penser while dispensing. This gap is added for each dispensation, generating a remainder in the dispenser that keeps increasing its quantity after some of these instructions. The remove waste high-level command empties the syringe. It is important to execute this command before the remainder and the quantity that will be suctioned exceed the dispenser maximum quantity.

5 Building XML Protocols

In an analysis of a virus like HPV, the number of Basic-Commands that the protocol can have is 400 per sample, approximately. Creating an XML protocol manually for an analysis with more than one sample could take an inconceivable amount of time. This is why protocols are not created manually by researchers; it would take many hours to create it and, due to the fact that it would have a lot of Basic-Commands, the human error possibility is very high. To solve this problem, a software tool was developed to generate the XML protocol from a template written in CSV format [12] which consists of very high-level instructions that will later be translated into basic and high-level commands. A CSV template can be written using more than 30 different instructions, allowing to generate protocols that describe the same functionality as those created by using only Basic and High-Level commands, but spending less time and effort and reducing the error probability. A CSV template for the HPV analysis has around 63 instructions and the time spent to create it is 30 min approximately.

Therefore, the researcher needs to create a generic CSV protocol template based on his/her experiments and validations. Then, after selecting which reagents and samples are going to be used in the protocol, the software tool reads the CSV file associated with the selected analysis and automatically converts it to a valid XML protocol, which can be understood and executed by the robot based on this information. As it is written in a generic way, the CSV template, once created, is useful for the analysis regardless of which reagents, samples or membranes are used, being completely independent from these values. However, XML protocols can still be built manually or even modified after being generated using a CSV for further control over it.

Basic and High-Level commands are programmed as classes and functions in C#, respectively. This means that adding new commands that were not described in Sect. 3 can be done just by creating a new class defining the command functionality and binding it to the main architecture class. This easy process allows a quick and effortless adaptation of this protocol system to new instruments or robot functionalities.

6 Results

The time that the analysis process takes to complete depends on two factors: first, the kind of analysis that will be executed and, second, the number of samples that will be analyzed. Figure 4 shows execution times for the HPV virus diagnosis protocol using 1, 2, 6, 12, 15 and 24 membranes. These six tests were

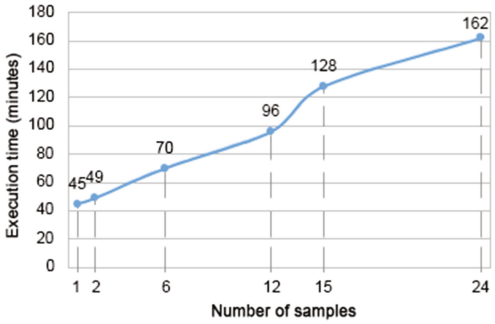

Fig. 4. Execution times of the HPV diagnosis protocol for 1, 2, 6, 12, 15 and 24 samples.

executed in HS24. As can be seen, time increases linearly except for the range between 12 and 15 samples, where the second reaction chamber (12 samples per reaction chamber) needs to be used, which is farther from the washing station and the hybridization buffer containers than the first one. Table 1 shows execution times for different number of samples, the number of instructions in the CSV file and the number of Basic Commands in the XML protocol. As it is an automated process without the need of human supervision, robots can be programmed with protocols during nighttime, increasing the number of tests per day and reducing human cost. The performance and robustness of the automatic HS24 system was validated by testing[10] low concentrations of the HPV genotypes included in the detection kit for HPV (5 copies for HPV 16 and HPV 18, 50–500 copies for the other genotypes).

Basically, this validation demonstrates the reproducibility of the results in positions 1 to 24 on the HS24 device and the reproducibility of the results for runs with different number of samples. Moreover, the system demonstrated a high sensitivity of detection for all the genotypes.

Table 1. Execution time, number of instructions and number of Basic Commands for 1, 2, 3, 6, 12, 15 and 24 samples.

Number of samples	Execution time (minutes)	Number of instructions	Number of Basic Commands
1	45	63	967
2	49	63	1161
6	70	63	2325
12	96	63	3877
15	128	63	5065
24	162	63	7587

[10] All tests were performed at Master Diagnostica.

The reproducibility of the results for runs with different number of samples was evaluated. Replicates for a positive sample containing a limiting copy number of several genotypes (50 GE) were loaded at different positions in the HS24 platform, according to four protocols: (1) Protocol for 2 samples (2 replicates), (2) protocol for 12 samples (3 replicates), (3) protocol for 15 samples (4 replicates), (4) protocol for 24 samples (6 replicates).

The results were automatically analyzed and all the samples were detected without differences among positions and protocols.

In order to verify the reproducibility of the protocol in different positions of the HS24 reaction chambers, four replicates for each of the 36 HPV genotypes detected by the method at a limiting copy number were loaded in different positions of the two reaction chambers in HS24, and protocols for 24 samples were performed. The results were automatically analyzed and showed 100% reproducibility for all the genotypes analyzed in the different positions of the reaction chambers using classical image processing techniques during instruments validation at Master Diagnostica Labs for around 2 k samples. A LeNet-5 CNN has been implemented in Caffe and a the under construction database is being created using images taken from membranes after applying HPV analysis. For MNIST experiments 60k training samples were needed to obtain 98% accuracy. We aim to obtain similar results with similar database length on these HPV images.

7 Conclusions

This manuscript presents a novel XML-based hierarchical protocol architecture for biomedical robots that allows to describe any execution step that a HPV diagnosis protocol could need. This architecture is already implemented in hybriSoft (hybriSpot 24 sample management software from VITRO), allowing the HPV DNA detection processing of up to 24 samples simultaneously in less than three hours, which is a novelty in terms of speed in the field of biomedical robotic systems.

As XML files are generated automatically using a generic CSV template of the protocol, regardless of the number of samples and the reagents that will be used, the user does not have to create a new protocol for the same virus diagnosis when using a different number of samples, reducing time and effort to zero in these cases. This also leads to having a minimum human error probability when creating the protocol. Adjusting and adapting specific steps of the protocol can be done by modifying a single XML or CSV file, which means a fast and effortless way to change the robot behavior.

Although in this manuscript we focused on HPV DNA detection, this protocol architecture allows the detection of other infections like sepsis, meningitis, etc. which are already implemented on the hybriSpot 24 and hybriSpot 12-Auto systems.

Acknowledgments. We would like to thank VITRO for their continuous support and help during this work and for lending us a hybriSpot 24 for protocol testing purposes.

This work is supported by FIDETIA (P055-12/E03). This work is supported by the Spanish government grant (with support from the European Regional Development Fund) COFNET (TEC2016-77785-P).

References

1. Bosch, F.X., Lorincz, A., Munoz, N., Meijer, C.J.L.M., Shah, K.V.: The causal relation between human papillomavirus and cervical cancer. J. Clin. Pathol. **55**, 244–265 (2002)
2. Walboomers, J.M., Jacobs, M.V., Manos, M.M., Bosch, F.X., Kummer, J.A., Shah, K.V., Munoz, N.: Human papillomavirus is a necessary cause of invasive cervical cancer worldwide. J. Pathol. **189**, 12–19 (1999)
3. Wong, A.A., Fuller, J., Pabbaraju, K., Wong, S., Zahariadis, G.: Comparison of the hybrid capture 2 and cobas 4800 tests for detection of high-risk human papillomavirus in specimens collected in PreservCyt medium. J. Clin. Microbiol. **50**, 25–29 (2012)
4. Abreu, A.L.P., Souza, R.P., Gimenes, F., Consolaro, M.E.L.: A review of methods for detect human Papillomavirus infection. Virol. J. **9**, 262 (2012), http://doi.org/10.1186/1743-422X-9-262
5. Bradski, G.: The opencv library. Doctor Dobbs J. **25**, 120–126 (2000)
6. Bray, T., Paoli, J., Sperberg-McQueen, C.M., Maler, E., Yergeau, F.: Extensible markup language (XML). World Wide Web J. **2**, 27–66 (1997)
7. Savchenko, Y.: Supercavitation-problems and perspectives. In: CAV 2001, pp. 1–8 (2001)
8. Luo, Y., Yang, H., Hu, Z.: Human limb motion real-time tracking based on CamShift for intelligent rehabilitation system. In: IEEE International Conference on Robotics and Biomimetics (ROBIO), pp. 343–348 (2009)
9. Pingxian, Y., Rong, G., Peng, G., Zhaoju, F.: Research on lip detection based on Opencv. In: International Conference on Transportation, Mechanical, and Electrical Engineering (TMEE), pp. 1465–1468 (2011)
10. Li, X., Han, G., Ma, G., Chen, Y.: A new method for detecting fatigue driving with camera based on OpenCV. In: International Conference on Wireless Communications and Signal Processing (WCSP), pp. 1–5 (2011)
11. Manaf, A. S., Sari, R. F.: Color recognition system with augmented reality concept and finger interaction: case study for color blind aid system. In: 9th International Conference on ICT and Knowledge Engineering, pp. 118–123 (2012)
12. Shafranovich, Y.: Common format and MIME type for comma-separated values (CSV) files (2005)
13. Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., Darrell, T.: Caffe: Convolutional Architecture for Fast Feature Embedding. arXiv preprint arXiv:1408.5093 (2014)