|  |
| --- |
| 1. Invention |
| Invention Title:A Vision Intelligence Framework for Mobile Devices |
| Lab:CSIC | Project Name:Computer Vision  | Project Code (PLM): RDR0116ZZ-10PG | Cost Center: C3400423 |
| Project Related HQ Person:      | Related HQ Project (if applicable):      | Project Funding HQ Business Unit:      |
| Check if urgent:   | Reason:Technology for 2017/2018 products |
| 2. Inventor(s) |
| Full Name(Last, First MI) | Home Address(Street Address, City, State, Zip) | Work Phone &Email Address | Citizenship(Country) | Empl. Status(Employee/Dispatcher/Contractor) |
| Antol, Stanislaw |       |       |       |       |
| Bendale, Abhijit |       |       |       |       |
| Gibbs, Simon |       |       |       |       |
| Jeon, Won |       |       |       |       |
| Kang, Hyun Jae |  |  |  |  |
| Kim, Jihee |  |  |  |  |
| Li, Robert |  |  |  |  |
| Liot, Anthony |  |  |  |  |
| Luo, Lu |  |  |  |  |
| Mistry, Pranav |  |  |  |  |
| 3. Conception of the Invention |
| a. Was the invention conceived in the United States?  | If not, in what country was the invention conceived?      |
| b. Date of first written description: July 2016  | Where can this description be found?SRA / Mountain View (CV shared folder, confluence: Maya space) |
| c. Was invention developed using non-Samsung (e.g., university, government) funding? | If so, what was the source of funding?      |
| 4. Construction of the Invention |
| Was a model or prototype made? | If so, when was it constructed and where can it be found?August 2016, SRA / Mountain View (stash server: arrow and maya repos) |
| 5. Use of the Invention |
| Are there specific plans for its use? | If so, when was it (or will it be) used?2017 |
| For what purpose will it be used? (e.g., project/product name, industry standard)New features for mobile camera and gallery applications. |
| 6. Disclosure/Sale of Invention (either past or near future) |
| a-1. Has invention been (or will it be) disclosed to others outside Samsung?  | If so, to whom was it (or will it be) disclosed? (e.g., vendor, conference, standards body)      |
| If so, when was it (or will it be) disclosed?      |
| a-2. Was the disclosure or will the disclosure be made under a Non-Disclosure Agreement?  | If so, when was it (or will it be) executed?      |
| b. Has Samsung actually sold (or will it be selling) any product using the invention?  | If so, when was (or will) the product be first sold?2017 |

|  |
| --- |
| 7. Related Invention Disclosures or Patent Applications By Inventors (briefly explain relation) |
| * CSI17-CV01 (Maya UI)
* CSI17-CV03 (Contextual DL)
* CSI17-CV05 (Hiearachical DL)
 |
| 8. Invention Overview |
| **a) Background of Invention**The camera is the most widely used application on mobile devices. While the primary role of this application is to capture photos and videos, with the addition of new AI and Computer Vision technology the camera is evolving into an intelligent tool that extracts information from the visual domain and provides rich visual context to other applications and services on the device. To perform this feat of visual analysis, the camera must be extended with image processing functionality based on deep learning and neural networks. |
| **b) Summary of Invention**The invention describes an system for extending the mobile camera with a set of vision intelligence components characterized by the following:* An open architecture engendering a component ecosystem with partners and third parties
* Autonomous vision intelligence components allowing their dynamic addition, removal, activation, deactivation or upgrade
* Small footprint in camera application providing easy integration
* Sandboxing of components to isolate the camera application from component failures
* Component load balancing providing flexibility in power optimization and resource utilization
* High-degree of modularization allowing components to benefit from new forms of HW acceleration (GPU, FPGA, special purpose neural net processing units) without change to camera application itself
* Support for both on-device and on-cloud components and hybrid architectures that combine on-device and on-cloud processing of visual data
 |

|  |
| --- |
| **c) Primary Value of Invention to Samsung**  |

|  |
| --- |
| **d) Closest Existing Technologies to Invention**The invention relies on DL (Deep Learning) technology – in particular deep neural networks used for image and video processing such as convolutional neural networks (CNNs). These networks are capable of a variety of image processing tasks that includes image classification and object detection.With the recent development of very large training sets, deep neural networks have become highly accurate and have approached or surpassed humans for some tasks. While training these networks requires considerable computing and storage resources, their deployment is much less demanding and current mobile devices are sufficiently powerful to run trained models in near real-time.Several companies (Google, Facebook, Microsoft and others) have built “engines” for running DL models on mobile devices. Optimizing DL engines and DL models for mobile deployment is an area of active research and development that includes technologies for model compression and specialized hardware for neural net computation.However this existing work, while providing building blocks, does not directly address how to harness new technologies in mobile applications. This invention explicitly addresses this issue. |
| **e) New Features of Invention** Briefly describe the key specific structural or functional features of your invention that differentiate it from the closest existing technologies identified in item 8(d).* Extension framework for mobile camera: rapid deployment.
* Component ecosystem: provisioning of deep learning models from community, vendor, user and partners/third parties
 |
| **f) Advantages/Disadvantages of Invention**Key advantages:* Simple integration with camera application.
* Sandboxed components, isolates failures.
* Component ecosystem.
* Modularized architecture providing load balancing opportunities for resource and power management
 |

|  |
| --- |
| 9. Detailed Description of Invention |
|  |

|  |
| --- |
| In one embodiment, as illustrated in Figure 3, the *vision intelligence framework* provides a set of APIs and software components helping developers enhance existing applications and build new applications that leverage the processing of and understanding of visual information. The framework encapsulates a particular vision task, such as object recognition or face detection, with a software abstraction called a *module*. Modules have easy-to-use interfaces for configuration and state management. Once a module is configured, an application typically registers listeners to receive results from the module. Modules may be configured for streaming operation, in which case live camera or recorded video frames are continuously passed to the module, or they may be configured for “single shot” operation, allowing a single camera frame or photo to be passed to the module. The mechanics of sending image data to modules is handled by the framework itself.In one embodiment, the framework groups modules into multiple families, such as *mDL*, *mSymbol*, *mFace*, *mTrack, mCloud* and *mGesture* – each family containing modules associated with a specific problem area, such as tracking, or a particular technology, such as Deep Learning.In one embodiment, modules run in their own containers and are highly autonomous. This has the benefits of reducing application launch time and also assuring that applications remain running should modules happen to fail. Since modules may have significant resource demands (memory, processing), sandboxing modules is important to preserve the integrity and reliability of critical applications such as the Camera or Gallery (offline storage of pictures taken from camera). In one embodiment, module isolation is achieved by running modules in separate processes in which case the framework provides applications with APIs for launching module processes and communicating with modules through IPC and shared memory.The vision intelligence framework also has the goal of allowing easy integration of modules with existing applications. To simplify integration and further isolate existing applications from modules, in one embodiment, the framework provides a plugin-mechanism referred to as *Layers*. A layer is a small embeddable application that handles communication with one or more modules, which is illustrated in Figure 1. In one embodiment, a layer enables a lightweight application by isolating Augmented Reality (AR) modules from a camera app. It helps selectively add/remove/change/run AR modules to/from the camera App. Such a layer enables easy integration of AR modules into the camera app. For example, for a camera app, an office view captured by the camera can be processed by object recognition module to recognize a furniture inside the office, or by a scene recognition module to recognize it is an office space. The layer enables such flexible operations. Because of their small size, layers have little impact on the resource footprint of the hosting application. Using layers, the application has no need to deal with modules directly and can easily turn on and off different modules.Figure 1 illustrates an exemplary application hosting a layer, which is responsible for interacting with one or more modules running in one or more module containers. In one embodiment, the one or more modules process visual input data (e.g., camera frames) from a source, and then provide visual results (e.g., identified objects) to the layer, or to the application directly.Modules provide visual processing tasks such as recognizing the types of objects present in a camera frame (classification), detecting where objects are located (localization, segmentation) or tracking objects over a series of frames. Generally speaking, modules have an internal structure that includes models, engines and an engine selector. As an example, the structure of the deep learning (DL) module is shown in Figure 2.In one embodiment, a user chooses which module will be used for a specific task or operation. In a second embodiment the application is responsible for selecting which modules to use.At the bottom of Figure 2, there is a set of models. In the case of the DL module, models are neural nets that have been trained for specific tasks. In one embodiment of the DL module, the models and the data needed for training these models may include:* Community models –models developed by the research community and made freely available
* Vendor models – models developed by the device vendors and only available to run on devices from that vendor, or devices licensed by that vendor
* User created models – models created directly from content provided by the user, an example would be a model trained using a user’s photo collection to identify friends or family members
* Third party / partner models –models developed by third parties and made available through partnership or licensing

In one embodiment, developers of the application controls which engine or engines are associated with the DL module. The choice may be based on requirements and/or parameters of a model. The DL Engine Selector may select the appropriate engine for the chosen engine or engines. In another embodiment, the engines are selected automatic by the vision intelligence framework. For example, each model may be supported by an engine, and one engine may support multiple models. The system may dynamically determine association between a model and an engine based on various runtime data. Such an embodiment removes the fixed association between an app and an engine so that the same app can run various engines, including 3rd party engines, based on information such as ontology, application specifics, contextual information surrounding the operation of the app, etc. For example, with a camera app, while the system understands it is expected to run object recognition, it may determine to run a poster recognition of an image present in the camera view, upon receipt of GPS data indicating that the user is at a movie theatre. For the same camera app, the system may determine, upon recognition of an animal, to further run another model to provide geographic or other data of the recognized animal, based on an ontology of models.I:\Patent\Invention Disclosures\CSIC\2017\Computer Vision\CSI17-CV02\maya-architecture-0.2.jpgFigure 3 |

|  |
| --- |
| 10. Alternative Embodiments of Invention |
| Describe any alternatives to the embodiment described in item 9. Drawings, diagrams and/or flowcharts can be helpful to illustrate your invention.  |

|  |
| --- |
|  |

|  |
| --- |
| 11. Signatures |
| **a) Inventors**I/we, the undersigned, are the sole and original inventor(s) of this invention. |
| Inventor 1 | Date |  | Inventor 2 | Date |
| Inventor 3 | Date |  | Inventor 4 | Date |
| **b) Witnesses**I/we, the undersigned, have reviewed this invention disclosure and understand its contents. |
| Witness 1 | Date |  | Witness 2 | Date |
| **c) SRA-SV Management**We, the undersigned, have reviewed and approve of this invention disclosure as to its scope and completeness. |
| Project Lead | Date |  | Lab Manager | Date |

Inventors (and witnesses) should initial and date each page of Invention Disclosure and any attachments. Submit completed Invention Disclosure to SRA-SV Patent Dept. If you have any questions, please contact Thomas George at (408) 544-5576 (Thomas.g@samsung.com), Jade Sche at (408) 544-5945 (j.sche@samsung.com), Justin Chang at (408) 544-5677 (justin.chang@samsung.com), Louisa Toy at (408) 544-5083 (louisa.toy@samsung.com), or Ruke Wang at (408) 544-5615 (ruke.wang@samsung.com).