GSOC 2018 ideas

Projects that we maintain:
- libva : https://github.com/01org/libva
- libva-utils: https://github.com/01org/libva-utils
- vaapi-intel-driver : https://github.com/01org/intel-vaapi-driver
- gstreamer-vaapi : https://cgit.freedesktop.org/gstreamer/gstreamer-vaapi/
- gstreamer-MediaSDK: https://cgit.freedesktop.org/gstreamer/gst-plugins-bad/tree/sys/msdk
- libyami: https://github.com/01org/libyami
- libyami-utils: https://github.com/01org/libyami-utils
- libxcam: https://github.com/01org/libxcam

List of Project Ideas:
libva/libva-utils
- Implement a simple HEVC(h265) and VP8 encoder
  We have similar h264 and vp9 encoders as a part of libva-utils. Add similar encoder applications for HEVC and VP8 codec too. H264/AVC encoder application code is here: https://github.com/01org/libva-utils/blob/master/encode/avcenc.c . Here is the vp9 encoder application: https://github.com/01org/libva-utils/blob/master/encode/vp9enc.c

    Difficulty: Medium
    Skill Required: C
    Optional Skills: Understanding of H265/HEVC and VP8 Codec
    Hardware requirement: Intel Skylake, Broxton, Kabylake or Coffelake microarchitecture based PC
    Possible Mentor: Sreerenj Balachandran <bsreerenj@gmail.com>
    IRC: sree_ on freenode/gstreamer
    Slack: sree on https://intel-media.slack.com/

libxcam:
- Improve Defog/Dehaze quality and performance. Design and tune algorithm based on Dark Channel Prior to improve image quality, especially on Halo removal and color balance. Also need to consider performance improvement based on OpenCL in IA platform.
   Difficulty: Medium
   Skill Required: C/C++, OpenCL
   Optional Skills: Familiar with image processing
   Hardware required: IvyBridge+ PC
   Possible Mentor: Wind Yuan <feng.yuan@intel.com>
   Slack: fengyuan on https://intel-media.slack.com/

- Enable HDR based on different exposure images.
   Investigate HDR algorithms based on 2 or 3 Low, (mid,) long exposure images into one clear image, Enable HDR feature into libxcam (https://github.com/01org/libxcam).Performance improvement based on Intel CPU/GPU also need to be considered.
   Difficulty: Medium
   Skill Required: C/C++/OpenCL
   Optional Skills: OpenCL/OpenCV/Image processing algorithm.
   Hardware Requirement: Intel Skylake+ based PC
   Possible Mentor: Wind Yuan <feng.yuan@intel.com>

- Supper resolution from low-resolution images.

Investigate supper resolution algorithms from low resolution images, This feature need to keep a clear edge. Implement it into libxcam (https://github.com/01org/libxcam). Parallel computing (OpenCL) for performance must be considered.

    Difficulty: Medium
    Skill Required: C/C++/OpenCL
    Optional Skills: OpenCL/Image processing algorithm.
    Hardware Requirement: Intel Skylake+ based PC
    Possible Mentor: Wind Yuan <feng.yuan@intel.com>

- 3D scene reconstruction by multi-cameras or multi-images.

Investigate algorithms for 3D scene reconstruction. This feature is based on multiple images from single/multiple cameras. And the process need consider at least camera calibration, depth-map generation, point cloud exaction and 3D rendering. OpenCV/OpenCL can be a good utility libs for image processing. Implementation should be in libxcam (https://github.com/01org/libxcam). Performance improvement is a plus.

        Difficulty: High
        Skill Required: C/C++/OpenCL/OpenCV
        Optional Skills: Familiar with Image processing / deep learning.
        Hardware Requirement: Intel Skylake+ based PC
        Possible Mentor: Wind Yuan <feng.yuan@intel.com>

- Enable camera online (auto) calibration without pattern checker.

Investigate camera auto-calibration (intrinsic / extrinsic parameters) algorithms from single or multiple cameras, usually it's based on still scene as background. Consider multiple cameras first. Implement it into libxcam (https://github.com/01org/libxcam).

    Difficulty: Medium
    Skill Required: C/C++
    Optional Skills: Image processing algorithm / OpenCV.
    Hardware Requirement: Intel CPU based PC
    Possible Mentor: Wind Yuan <feng.yuan@intel.com>

- Lane departure warning (LDW) based on image processing.

Investigate LDW algorithms of signal image processing, must consider accuracy improvement. This may need line detection and deep-learning based algorithms. Try to implement it into libxcam (https://github.com/01org/libxcam).

    Difficulty: High
    Skill Required: C/C++/OpenCV
    Optional Skills: Image processing / deep learning algorithms.
    Hardware Requirement: Intel Skylake+ based PC
    Possible Mentor: Wind Yuan <feng.yuan@intel.com>

gstreamer-vaapi:

* Add Scalable Video Encoding (SVC) support in H264 and VP8 encoder

  * Difficulty: High
  * Skills Required: C
  * Useful skills: video processing
  * Hardware/Software required: Intel CPU Sandybridge or newer

* Description: Finish the implementation of SVC support in the gstreamer-vaapi H264 encoder. The initial implementation needs to be cleaned up and tested,
   which is only supporting temporal svc. Implement the spatial svc encoding and adding bit rate control over different temporal layers are the major tasks involved.
   Similar svc support needs to be enabled for VP8 encoder too.

   https://bugzilla.gnome.org/show_bug.cgi?id=725536
   https://cgit.freedesktop.org/~sree/gstreamer-vaapi/log/?h=svc-enc

   Possible mentors:
   Sreerenj Balachandran (sree_ on IRC/gnome), bsreerenj@gmail.com/sreerenjb@gnome.org
   Víctor Jáquez (ceyusa on IRC/gnome), vjaquez@igalia.com

- Add support for inter-view prediction in h264 MVC encoder

   * Difficulty: High
   * Skills Required: C
   * Useful skills: video processing
   * Hardware/Software required: Intel CPU Skylake or newer
   * Description: Multiview video contains a large amount of inter-view statistical dependencies, since all cameras capture the same scene from
    different viewpoints. Therefore, combined temporal and inter-view prediction is important for efficient MVC encoding. A frame from a certain camera can be predicted not only from temporally related frames from the same camera,
   but also from the frames of neighboring cameras. These interdependencies can be used for efficient prediction.
   https://bugzilla.gnome.org/show_bug.cgi?id=745096

   Possible mentors:
   Sreerenj Balachandran (sree_ on IRC/gnome) : sreerenjb@gnome.org/bsreerenj@gmail.com
   Víctor Jáquez (ceyusa on IRC/gnome): vjaquez@igalia.com

   * Bug fixing and feature development

   * Description: Help us to reduce the bug count: we have 100+ open bugzilla
items: https://bugzilla.gnome.org/buglist.cgi?bug_status=UNCONFIRMED&bug_status...
Not all of them are bugs, many are feature request from customers.
   * Difficulty: Medium
   * Skills Required: C
   * Useful skills: Some understanding of Multimedia, Video in general
   * Possible mentors:
      Hyunjun Ko <hko@igalia.com >
      Sreerenj Balachandran <bsreerenj@gmail.com, sreerenj.balachandran@intel.com>
      Victor Jaquez <vjaquez@igalia.com>

gstreamer-MediaSDK Plugins:

 * Bug fixing and performance improving

* Description: Meida SDK is Intel's Software library for media acceleration : https://software.intel.com/en-us/media-sdk
We are maintaining the GStreamer plugins for MediaSDK: https://cgit.freedesktop.org/gstreamer/gst-plugins-bad/tree/sys/msdk
This is still in the early stages of development and there is plenty of room for bug fixing, feature development, and performance improvement. Please let us know if you are interested to work on this.
* Difficulty: Medium
* Skills Required: C
* Useful skills: Some understanding of Multimedia, Video in general
* Possible mentors:
  Hyunjun Ko <hko@igalia.com >
  Sreerenj Balachandran <bsreerenj@gmail.com, sreerenj.balachandran@intel.com>
  Victor Jaquez <vjaquez@igalia.com>

libyami:
- Add temporal and spatial encoding support in VP9 Encoder
  Currently we have h264 temporal encoder at (https://github.com/01org/libyami/commit/ea0b5fd79715d0a154b79319d113cf50...) and vp9 encoder (https://github.com/01org/libyami/blob/apache/encoder/vaapiencoder_vp9.cpp)
  Difficulty: Medium
  Skill Required: C++
  Optional Skills: Understanding of VP9 codec
  Hardware required: Intel KabyLake+ PC
  Possible Mentor: Xu Guangxin <Guagnxin.Xu@gmail.com>
  Slack: guangxin on https://intel-media.slack.com/

- Add runtime check for all hardware supported features
  We have compile time capability query function here(https://github.com/01org/libyami/blob/apache/decoder/vaapidecoder_host.c...). It will report supported decoder/encoder. Extent the API and query driver capability using vaQueryConfigEntrypoints, and report driver supported decoder/encoder.
  Difficulty: Medium
  Skill Required: C++
  Optional Skills: Understanding of VAAPI
  Hardware required: Intel SandyBridge+ PC
  Possible Mentor: Xu Guangxin <Guagnxin.Xu@gmail.com>
  Slack: guangxin on https://intel-media.slack.com/

- Add C APIs for Video Post Processing
  We have C API for decoder(https://github.com/01org/libyami/blob/apache/capi/VideoDecoderCapi.h) and encoder (https://github.com/01org/libyami/blob/apache/capi/VideoEncoderCapi.h) C API P. Similarly Add video post process C API.
  Difficulty: Easy to Medium
  Skill Required: C++
  Hardware required: IvyBridge+ PC
  Possible Mentor: Xu Guangxin <Guagnxin.Xu@gmail.com>
  Slack: guangxin on https://intel-media.slack.com/

libyami-utils:

- Add feature to share buffers between camera and encoders
  We can read yuv frames from the camera (https://github.com/01org/libyami-utils/blob/master/tests/encodeInputCame...) but it will copy yuv data many times. Add zero copy functions to the camera input.
  Difficulty: Medium
  Skill Required: C++
  Optional Skills: Understanding of V4L2
  Possible Mentor: Xu Guangxin <Guagnxin.Xu@gmail.com>
  Slack: guangxin on https://intel-media.slack.com/

- Add C API demo application for video postprocessing and encoder
  We have decoder capi test application(https://github.com/01org/libyami-utils/blob/master/tests/vppinputdecodec...). Add vpp and encoder C API test application.
  Difficulty: Easy to Medium
  Skill Required: C++
  Optional Skills:
  Possible Mentor: Xu Guangxin <Guagnxin.Xu@gmail.com>
  Slack: guangxin on https://intel-media.slack.com/

General Ideas (Add VA-API based Hardware Acceleration in other open source projects):

GCC support for xtensa SIMD intrinsics
Description: GCC currently supports SIMD intrinsics for x86, ARM and other CPU architectures but does not yet support SIMD intrinsics for the Xtensa architecture used on numerous low cost developer boards and found in all Intel SoCs and PCH's from Baytrail. Xtensa intrinsics are currently only supported by proprietary compilers that are a barrier to community development. Intrinsic support can give performance improvements of 25x or greater depending on use case.
Difficulty level: Medium. Prior art exists in gcc codebase for x86 and ARM. Intrinsic C headers already exist for Xtensa defining each intrinsic instruction. Binutils and GDB already have support for all Xtensa SIMD instructions used by the intrinsics.
Skil required: C coding, interest in compilers, SIMD, RISC architecture
Possible mentors:
Liam Girdwood <liam.r.girdwood@linux.intel.com>
Matthew Atwood <matthew.s.atwood@intel.com>

Chromium :Add VAAPI-based HEVC encoder on chromium
Add HEVC encoder support on chromium project. (Linux/Chrome OS)
Difficulty: Difficult
Skill Required: C++
Optional Skills: knowledge about HEVC spec
Possible Mentor: Qu Pengfei <Pengfei.Qu@intel.com>
Slack: Pengfei on https://intel-media.slack.com/

Chromium:Add DXVA2-based HEVC decoder on chromium
Add DXVA2 HEVC decoder support on chromium project.(Windows OS)
Difficulty: Difficult
Skill Required: C++
Optional Skills: knowledge about HEVC spec and DXVA2
Possible Mentor: Qu Pengfei <Pengfei.Qu@intel.com>
Slack: Pengfei on https://intel-media.slack.com/

GStreamer:Add VMAF filter as plugin in the GStreamer
VMAF(https://github.com/Netflix/vmaf) is the metric tool and perceive the video quality subjectively. It is similar to SSIM/PSNR. FFmpeg has this as a filter by now. Integrate VMAF as the GStreamer-Plugin.

Difficulty: Medium
Skill Required: C/C++
Optional Skills: knowledge about GStreamer
Possible Mentor: Qu Pengfei <Pengfei.Qu@intel.com>
Sreerenj Balachandran <sreerenj.balachandran@intel.com>
Victor Jaquez <vjaquez@igalia.com>
Hyunjun Ko <hko@igalia.com >