Karlsruhe Institute of Technology

Camera Abstraction

Matthias Vogelgesang – *matthias.vogelgesang@ipe.fzk.de*

Institut für Prozessdatenverarbeitung und Elektronik

Goals

- Unified camera abstraction layer for cameras with different access interfaces (CameraLink, PCIe, . . .)
- Full support for Linux and 64-bit environments
- Support for pco edge, photon focus, Photron, IPE camera . . .
- Faster (and cheaper) tango server development for these cameras

Implementation

- Lightweight library `libuca` written in C
- No dependencies except for camera and frame grabber SDKs where needed
- Run-time detection of camera type
- Code at `bzr+ssh://user@ufo.kit.edu/vogelgesang/libuca`

Initialization

```
struct uca_t *uca = uca_init();
struct uca_camera_t *cam = uca->cameras;

uint32_t val = 10000;
cam->set_property(cam, UCA_PROP_EXPOSURE, &val);
uint32_t width, height;
cam->get_property(cam, UCA_PROP_WIDTH, &width);
cam->get_property(cam, UCA_PROP_HEIGHT, &height);
```

Recording

```
uca_cam_alloc(cam, 10);
uint16_t *buffer = (uint16_t *) malloc(width*height*2);
cam->start_recording(cam);
cam->grab(cam, (char *) buffer);
cam->stop_recording(cam);
uca_destroy(uca);

FILE *fp = fopen("out.raw", "wb");
fwrite(buffer, width*height, pixel_size, fp);
fclose(fp);
```

Simple UI in less than 400 LOC