# Table of Contents



# written @ HA 06/2014

```
clear all;
clc;
disp('MeasureSensorPos00.m running...');
workspace;
% Wechsle Pfad wenn [...]
if(~isdeployed)
    cd(fileparts(which(mfilename)));
end
```

```
MeasureSensorPos00.m running...
```

# Timer Klasse für Wartezeiten

```
T = timer('TimerFcn',@(~,~)disp('Fired.'),'StartDelay',1);%init timer
start(T);%Starte Timer
```

# Welche Hardeware ist vorhanden, Adapter

```
imaqreset;%Alle Image Aquisition Klassen schließen
wait(T);
info = imaqhwinfo('gige');%Welche Image Aquisition Hardware? Adapter GiGe
dev_info =info.DeviceInfo;
dev_info_Device = dev_info(1).DeviceName;% Kamera Name z.B eco415
```

```
Fired.
```

# Videoobjekt, Schnittstelle zwischen MATLAB und Kamera

```
vid = videoinput('gige', dev_info_Device, 'Mono8');
vid.FramesPerTrigger = 1;
```

# Aktuelle VideoSource Klasse und Kamera Features

```
src = getselectedsource(vid);
get(src);% Eigenschaften der Kamera zeigen
```

```
            General Settings:
              Parent = [1x1 videoinput]
              Selected = on
              SourceName = input1
              Tag = [0x0 string]
              Type = videosource

            Device Specific Properties:
              AcquisitionFrameRate = 5
              AcquisitionStartTriggerActivation = RisingEdge
              AcquisitionStartTriggerDelay = 0
              AcquisitionStartTriggerMode = Off
              AcquisitionStartTriggerSource = Software
              AllBlackLevelRaw = 10
              AllGain = 0
              AllGainAuto = Off
              AllGainAutoLevel = 100
              DeviceID = 9852
              DeviceManufacturerInfo = Build 65 Firmware Nov  5 2010 13:40:13
              DeviceModelName = eco415MVGE_r2
              DeviceScanType = Areascan
              DeviceUserID = [0x0 string]
              DeviceVendorName = SVS-VISTEK GmbH
              DeviceVersion = 1.4.65
              ExposureMode = Timed
              ExposureTime = 20000
              FrameStartTriggerActivation = RisingEdge
              FrameStartTriggerDelay = 0
              FrameStartTriggerMode = Off
              FrameStartTriggerSource = Software
              Line1LineInverter = False
              Line1LineMode = Output
              Line1LineSource = Off
              Line1LineStatus = True
              Line2LineInverter = False
              Line2LineMode = Output
              Line2LineSource = Off
              Line2LineStatus = False
              Line3LineInverter = False
              Line3LineMode = Output
              Line3LineSource = Off
              Line3LineStatus = False
              Line5LineInverter = False
              Line5LineMode = Input
              Line5LineSource = Off
              Line5LineStatus = False
```

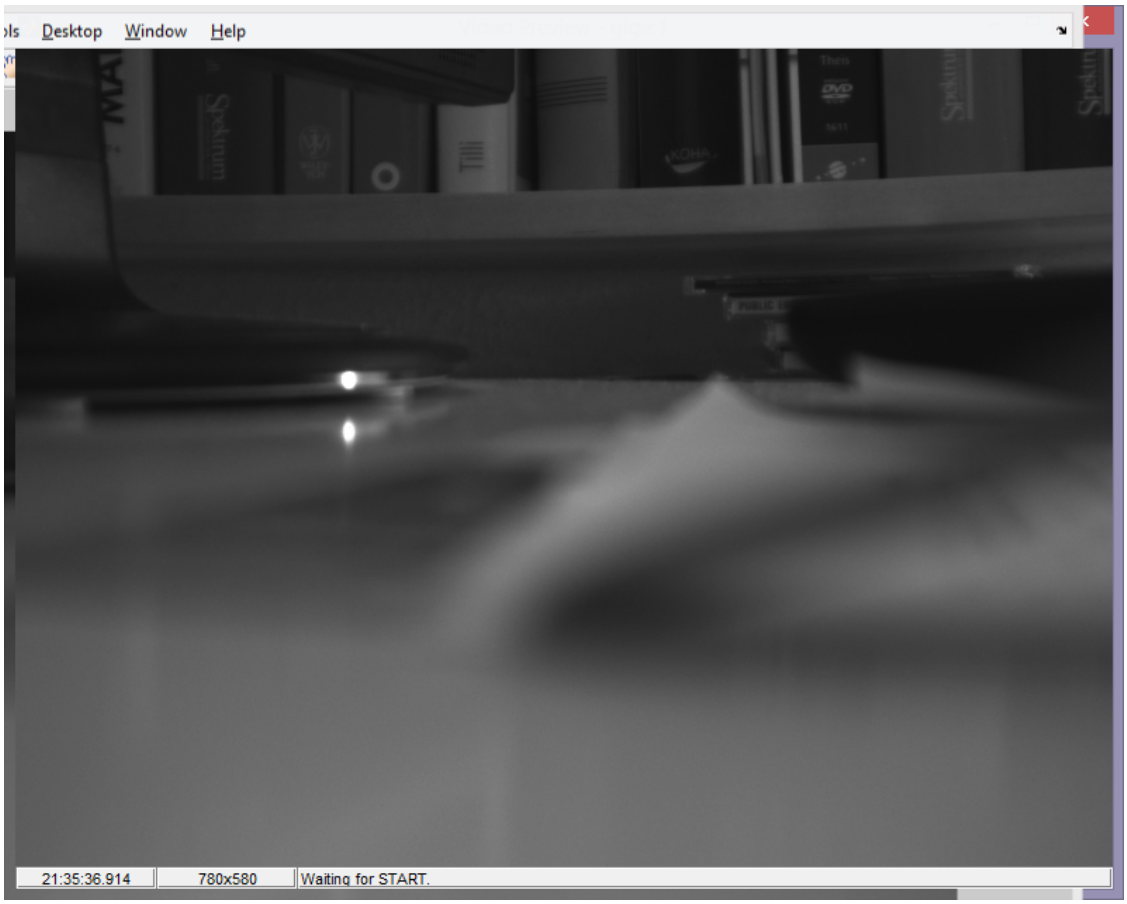
```
Line6LineInverter = False
Line6LineMode = Input
Line6LineSource = Off
Line6LineStatus = False
Line7LineInverter = False
Line7LineMode = Input
Line7LineSource = Off
Line7LineStatus = True
PacketDelay = 0
PacketSize = 1500
PixelColorFilter = None
PWMChange0 = 0
PWMChange1 = 0
PWMEnable = False
PWMMax = 66666
ReadoutControl = disable
ReadoutDelay = 1000
StrobeDelay = 0
StrobeDuration = 0
StrobePolarity = positive
TemperaturSensor = 43
TimestampTickFrequency = 66666666
TriggerDebounceDelay = 0
UserOutput1UserOutputValue = False
UserOutput2UserOutputValue = False
UserOutput3UserOutputValue = False
```

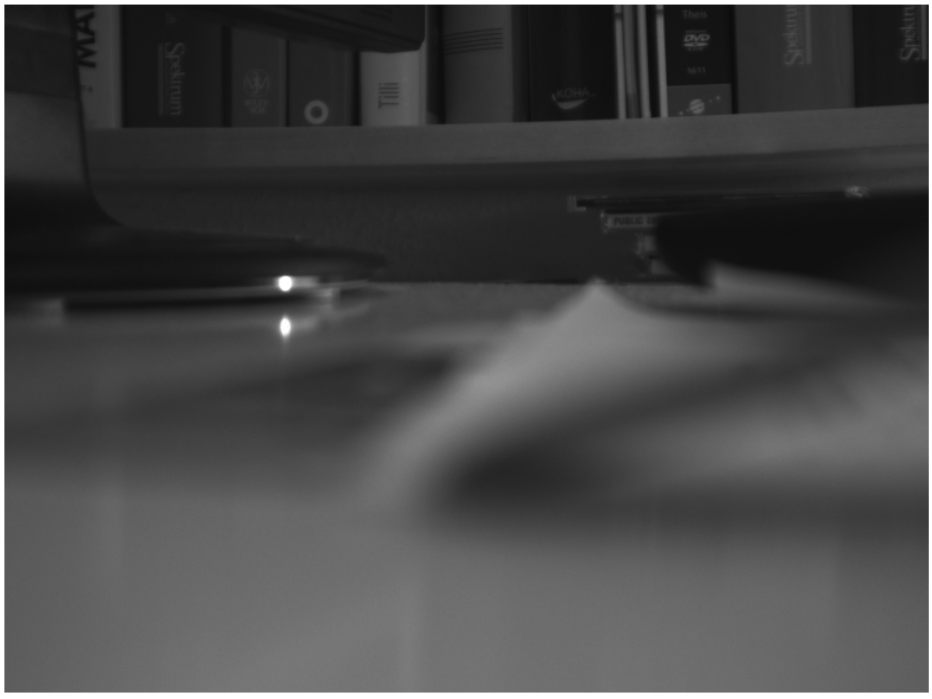
# KameraParameter lesen/schreiben/einstellen

```
src.ExposureTime = 20000;%Belichtungszeit 20000µsec
src.AcquisitionFrameRate = 5;%[Hz]
temperatur = get(src,'TemperaturSensor');%Lese Sensortemperatur
wait(T);
```

# Starte Image Aquisition und hole Bilddaten

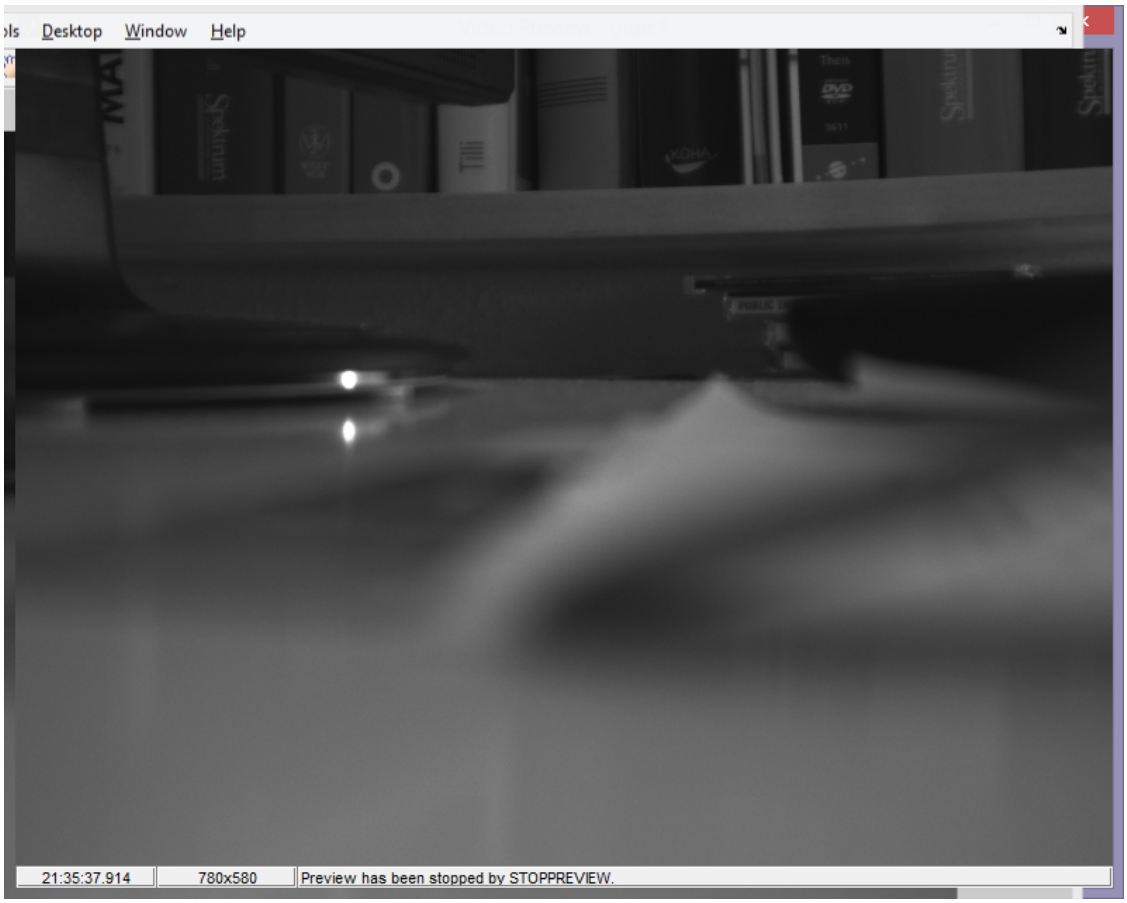
```
preview(vid);%Live Bilder von der Kamera
wait(T);% T = 1sec.
start(vid);%Starte Image Aquisition und speichere Device Konfiguration
im1 = getdata(vid);%Speichern des letzten Bildes. Achtung: FramesPerTrigger!
figure, imshow(im1);%zeige mir das Bilde im Fenster
bool = islogging(vid);
wait(vid);
```


ols Desktop Window Help
21:35:36.914
780x580
Waiting for START.



# Starte Sequenz mpeg

```
%writeObj = VideoWriter('Movie01.mp4''MPEG-4');
% open(writeObj);
% writeVideo(writeObj,im1);
stoppreview(vid);%keine online Vorshow
stop(vid);%Stop Image Aqusition
```

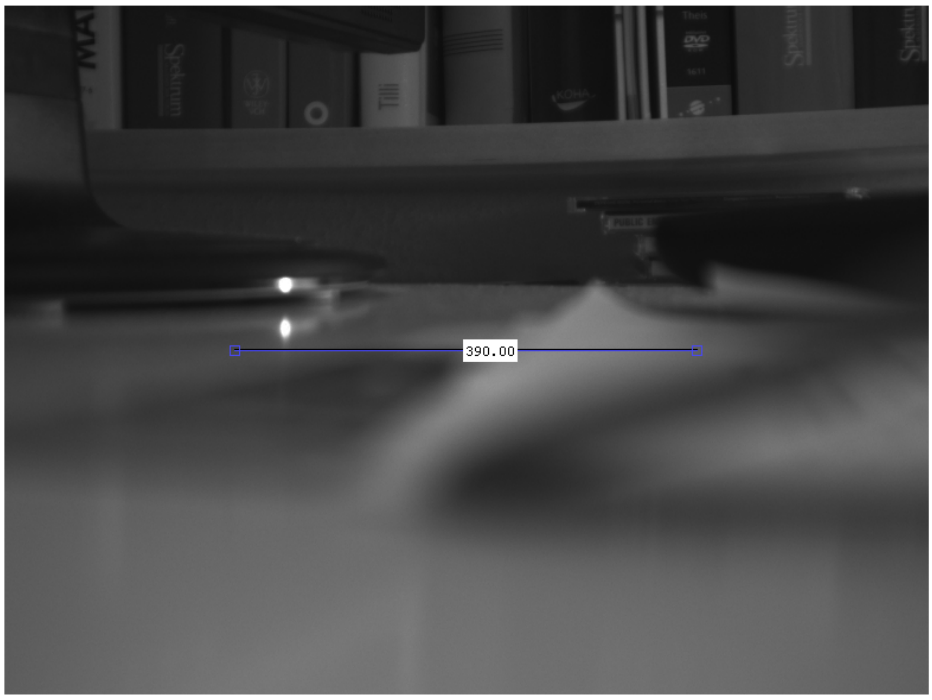



# Bilddaten speichern und Bild anzeigen

```
%save('Measure00.mat', 'im1');
%figure, imshow(im1);
```

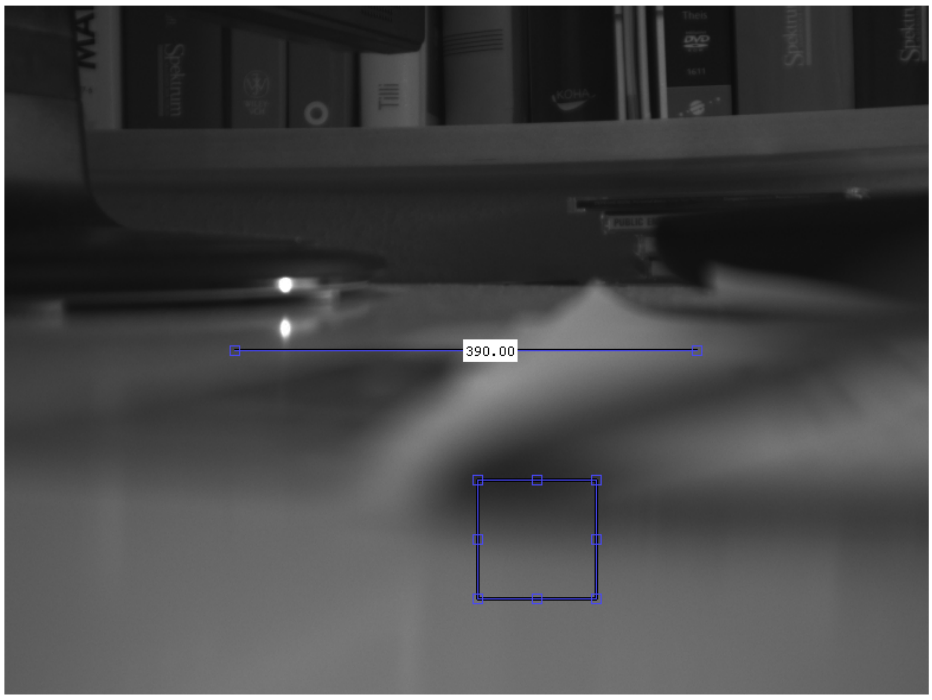
# Feature für Abstandsmessung in Figure

```
h = imdistline(gca);%Das Entfernungtool:Eine ziehbar, veränderbar Linie,die den Ab
api = iptgetapi(h);%Get Application Programmer Interface (API) for handle
fcn = makeConstrainToRectFcn('imline',...
                             get(gca,'XLim'),get(gca,'YLim'));%Create rectangular
api.setDragConstraintFcn(fcn);% API Funktion
```




# Feature Rechteck einblende in Figure

```
r = imrect(gca, [400 400 100 100]);
addNewPositionCallback(r,@(p) title(matstr(p,3)));
fcn = makeConstrainToRectFcn('imrect',...
                             get(gca,'XLim'),get(gca,'YLim'));
setPositionConstraintFcn(r,fcn);
```




# Videoobjekt schließen, Timer schließen

```
%close(writeObj);
delete(T); %Delete timer
delete(vid); % Video Klassefreigeben
clear vid;
clear all;% Workspace bereinigen
clc;%Kommandofenster bereinigen
```

*Published with MATLAB® R2013b*