

Video surface renderer objects initialization

By default, a surface renderer objects for video publishing and playback are initialized automatically. But in some Android versions an OpenGL graphics context should be explicitly set in this case, otherwise an application will crash during renderer initialization with `android.opengl.GLException`, for example

```
2024-06-13 15:57:47.006 32017-646/com.flashphoner.wcsexample.screencapture
I/EglBase14Impl: Using OpenGL ES version 2
2024-06-13 15:57:47.013 32017-646/com.flashphoner.wcsexample.screencapture
E/SurfaceTextureHelper: CaptureThread create failure
2024-06-13 15:57:47.013 32017-646/com.flashphoner.wcsexample.screencapture
E/SurfaceTextureHelper: android.opengl.GLException: Failed to create EGL
context: 0x3000
2024-06-13 15:57:47.015 32017-646/com.flashphoner.wcsexample.screencapture
E/SurfaceTextureHelper: [1/2] android.opengl.GLException: Failed to create
EGL context: 0x3000
    at org.webrtc.EglBase14Impl.createEglContext(EglBase14Impl.java:336)
    at org.webrtc.EglBase14Impl.-$$Nest$smcreateEglContext(Unknown
Source:0)
    at org.webrtc.EglBase14Impl$EglConnection.<init>
(EglBase14Impl.java:66)
    at org.webrtc.EglBase14Impl.<init>(EglBase14Impl.java:142)
    at org.webrtc.EglBase.createEgl14(EglBase.java:263)
    at org.webrtc.EglBase.create(EglBase.java:206)
    at org.webrtc.SurfaceTextureHelper.<init>
(SurfaceTextureHelper.java:187)
    at org.webrtc.SurfaceTextureHelper.<init>(Unknown Source:0)
    at
org.webrtc.SurfaceTextureHelper$1.call(SurfaceTextureHelper.java:75)
    at
org.webrtc.SurfaceTextureHelper$1.call(SurfaceTextureHelper.java:70)
    at org.webrtc.ThreadUtils$3.run(ThreadUtils.java:173)
    at android.os.Handler.handleCallback(Handler.java:874)
    at android.os.Handler.dispatchMessage(Handler.java:100)
    at android.os.Looper.loop(Looper.java:198)
    at android.os.HandlerThread
```

Since Android SDK build [1.1.0.65](#) a surface renderers may be initialized explicitly using application base OpenGL context `Flashphoner.eglBaseContext` to prevent a such exceptions.

To do this, automatic renderers initialization should be disabled when session is created

code

```
sessionOptions = new SessionOptions(mWcsUrlView.getText().toString());
sessionOptions.setLocalRenderer(localRender);
sessionOptions.setRemoteRenderer(remoteRender);
sessionOptions.setAutoInitRenderers(false);
```

```
...  
session = Flashphoner.createSession(sessionOptions);
```

and the surface renderers should be initialized explicitly

code

```
localRender = (SurfaceViewRender) findViewById(R.id.local_video_view);  
remoteRender = (SurfaceViewRender) findViewById(R.id.remote_video_view);  
...  
localRender.init(Flashphoner.eglBaseContext, null);  
remoteRender.init(Flashphoner.eglBaseContext, null);
```