```
//A real world race condition and critical section.  This is one of the problems I
worked on over the summer.  An OpenCV cameraView base that is not quite
threadsafe.  Where are the race conditions and critical sections?
//Broken
if (bmpValid && mCacheBitmap != null) {

      if (canvas != null)
            canvas.drawColor(0, android.graphics.PorterDuff.Mode.CLEAR);




            boolean bRotate = (userRotation !=0);

            //lets lock and rotate if necessary
            if (bRotate) {
               canvas.save();
               canvas.rotate(userRotation, ...);
            }
            if (mScale != 0) {
               canvas.drawBitmap(mCacheBitmap, new Rect((int)(canvas.getWidth(),..
            } else {
               canvas.drawBitmap(mCacheBitmap, new Rect(0,0,mCacheBitmap ..
            }
            if (mFpsMeter != null) {
                mFpsMeter.measure();
                mFpsMeter.draw(canvas, 20, 30);
            }
            //restore the canvas
            if (bRotate)
                canvas.restore();

            }
```

FIXED

```
if (bmpValid && mCacheBitmap != null) {
      Canvas canvas = getHolder().lockCanvas();  //aquire lock
      if (canvas != null) {
            canvas.drawColor(0, android.graphics.PorterDuff.Mode.CLEAR);
            //there is a race condition where you can change userRotation
            //in another thread and cause the the dreaded
            //'java.lang.IllegalStateException: Underflow in restore – more
            //restores than saves' happens because you change userRotation between
            //saving and restoring
            boolean bRotate = (userRotation !=0);    //userRotation is global

            //lets lock and rotate if necessary
            if (bRotate) {
               canvas.save();
               canvas.rotate(userRotation, ...);
            }
            if (mScale != 0) {
               canvas.drawBitmap(mCacheBitmap, new Rect((int)(canvas.getWidth(),..
            } else {
               canvas.drawBitmap(mCacheBitmap, new Rect(0,0,mCacheBitmap ..
            }
            if (mFpsMeter != null) {
                mFpsMeter.measure();
                mFpsMeter.draw(canvas, 20, 30);
            }
            //restore the canvas
            if (bRotate)
                canvas.restore();
                getHolder().unlockCanvasAndPost(canvas);   //release lock
            }
```