Using RTCMTLVideoView to display a video

Since build 2.6.6 it is possible to use `RTCMTLVideoView` to display a video, if device supports this view.

Use TwoWayStreaming example code to analyze automatic view selection to display a local (publishing) video:

1. Detect if Metal View is supported by device
   code

   ```
       id<MTLDevice> localDevice;
   #ifdef __aarch64__
       localDevice = MTLCreateSystemDefaultDevice();
       ...
   #endif
   ```

2. If `MTLCreateSystemDefaultDevice()` returns not `nil`, create `RTCMTLVideoView` object
   code

   ```
       ...
   #ifdef __aarch64__
       localDevice = MTLCreateSystemDefaultDevice();
       if (localDevice) {
           RTCMTLVideoView *localView = [[RTCMTLVideoView alloc] init];
           localView.delegate = self;
           _localDisplay = localView;
       }
   #endif
       ...
   ```

3. If `MTLCreateSystemDefaultDevice()` returns `nil`, create `RTCEAGLVideoView` object code

   ```
       ...
       if (!localDevice) {
           RTCEAGLVideoView *localView = [[RTCEAGLVideoView alloc] init];
           localView.delegate = self;
           _localDisplay = localView;
       }
       ...
   ```

Full code example for local video

```
    id<MTLDevice> localDevice;
#ifdef __aarch64__
```

```
    localDevice = MTLCreateSystemDefaultDevice();
    if (localDevice) {
        RTCMTLVideoView *localView = [[RTCMTLVideoView alloc] init];
        localView.delegate = self;
        _localDisplay = localView;
    }
#endif
    if (!localDevice) {
        RTCEAGLVideoView *localView = [[RTCEAGLVideoView alloc] init];
        localView.delegate = self;
        _localDisplay = localView;
    }
    _localDisplay.translatesAutoresizingMaskIntoConstraints = NO;
```