NuiTracker API (July 1st, 2014)

For readability, classes and methods that contribute to the library use "Hands" and "Bones" in their names when referring to "hand tracking" and "skeletal tracking" functionality to distinguish them from "Hand"/"User"/"Skeleton" used in NiTE naming scheme. Accessors will work as expected, ie. NuiTracker.getUserTracker() will return a NiTE UserTracker.

nite2.apps

- **TorsoPrinter**: Show tracked torso coordinates until Enter is pressed.
  - **main(String[] args)**
- **BonesAndHandsViewer**: Views both input methods (skeleton / hand) simultaneously.
  - **main(String[] args)**

nite2.basic

- **HandsListener (Interface)**: Any class that wants hand data should implement this interface and add itself to NuiTracker's hands listeners.
  - **onNewHandFrame(HandTrackerFrameRef frame)**
- **BonesListener (Interface)**: Any class that wants skeleton data should implement this interface and add itself to NuiTracker's bones listeners.
  - **onNewBonesFrame(UserTrackerFrameRef frame)**
- **NuiTracker**: Tracker for NUI features (implements HandListener and SkeletonListener from the NiTE library).
  - **addHandsListener(HandsListener listener)**
  - **addBonesListener(BonesListener listener)**
  - **getBufferedImage(): BufferedImage**
  - **getHandFrame(): HandTrackerFrameRef**
  - **getUserFrame(): UserTrackerFrameRef**
  - **getHands(): List<HandData>**
    - Get hand tracking data - see HandsRenderer for example of use.
  - **getHandTracker(): HandTracker**
  - **getBones(): List<UserData>**
    - Get skeleton data - see BonesRenderer or TorsoPrinter for example of use.
  - **getUserTracker(): UserTracker**
  - **onNewFrame(HandTracker ht)**
  - **onNewFrame(UserTracker ut)**
- **Utilities**: Some general utility methods
  - **distance3d(Point3D from, Point3D to): double**

nite2.gestures

- **JointMetrics**: Static methods for getting skeleton measurements etc.
  - **jointToJointDistance(UserData user, JointType from, JointType to): double**
    - Returns euclidian distance in space between two skeletal joints, eg. JointType.RIGHT_HAND, JointType.NECK etc. in millimeters
  - **elbowHandXOffset(UserData user, JointMetrics.Side side): double**
    - Returns the X axis distance in millimeters between elbow and hand.

- **JointMetrics.Side**: Side.LEFT or Side.RIGHT
- **Poses**: Methods to detect skeletal poses based on joint positions, eg. "hands above neck" etc.
  - **dorkyClick(UserData user): boolean**
    * Returns true if hands are above the neck and distance between the hands was less than 150 mm.
  - **handsAboveNeck(UserData user): boolean**
    * Returns true if a user's both hands are above the neck.

nite2.gui

- **GenericWindow**: Generic frame for displaying graphics
  - **run()**

nite2.gui.rendering

- **HandsRenderer**: Draw tracked hands (red rectangles when tracking) on top of depth image.
  - **onNewHandsFrame(HandTrackerFrameRef frame)**
  - **paint(Graphics g)**
    * Draw depth image and tracked hands.
- **BonesRenderer**: Draw stick characters from skeleton data on top of depth image.
  - **onNewBonesFrame(UserTrackerFrameRef frame)**
  - **paint(Graphics g)**
    * Draw depth image and skeletons.

nite2.gui.visualization

- **Visualization (Interface)**: Interface for classes that provide some sort of graphical presentation of the sensor data. A visualization will take a NuiTracker in its constructor to start listening to events.
  - **show()**
- **HandsVisualization**: Hand tracker visualization window
  - **show()**
- **BonesVisualization**: Skeleton tracker visualization window
  - **show()**