Abstract: The present invention includes provides secure, instant, and anonymous connections between two devices. The invention pairs a "cap" device with a capacitive touchscreen to a "cam" device with a camera sensor. For example, typical smartphones and tablets can be paired with each other, and these devices can be paired to even larger touchscreens, such as smart whiteboards and touchscreen monitors. The invention uses the cap device’s touchscreen to detect and track the cam device, and displays color-modulated pairing data directly underneath the camera once the camera is touching the screen. The pairing data is used as configuration data for a bidirectional link, such as an ad-hoc WiFi or Bluetooth link. These links are established without requiring user configuration. As such, the present invention provides a unidirectional communication mechanism from the touchscreen to the camera, which is used to bootstrap a full bidirectional, high-speed link.
System and Process for Enabling Secure, Instant, and Anonymous Communication Between Physically Touching Devices

Related Applications


Background

[0002] Recent years have seen an incredible proliferation in "smart" personal devices, such as wearables, smartphones and tablets. Larger displays, such as TV-sized touchscreens and smart whiteboards, are also becoming commonplace. While many such devices are reasonably smart on their own, cross-device interaction holds much promise for making devices even more capable and intelligent.

[0003] Cross-device pairing is commonly implemented using a wireless technology, such as WiFi or Bluetooth. Today, pairing devices is often cumbersome, requiring users to manually key in connection parameters (device or network) and security parameters (passphrase, PIN or pairing code). Though short-range NFC avoids many of these issues, as a wireless communication system, it is still subject to interception and man-in-the-middle attacks. Further, it is generally single device to single device, and lacks rich metadata such as spatial position of devices.

[0004] An ideal pairing solution would have the following properties:
[0005] **Zero-configuration**. Pairing should only use information already known to both devices.

[0006] **Rapid**. Pairing should be fast, ideally less than a few seconds from the start of the user pairing action to the establishment of a working connection.

[0007] **Multi-device**. Many devices should be able to pair simultaneously and continuously, each with their own secure and unique connection.

[0008] **Anonymous**. The pairing process itself should be anonymous. While the devices may later choose to exchange private data, an anonymous pairing procedure enables many more applications.

[0009] **Explicit**. The initial pairing should be explicitly initiated by a user action, so that user intent is clear.

[0010] **Secure**. The pairing should be difficult for attackers to intercept, tamper with, or forge.

[0011] **Targeted**. Pairing at a device level is fairly coarse, ideally, pairing should provide fine-grained additional information, such as a desired function or onscreen item.

[0012] **Rich Metadata**. Rich applications are enabled by rich sensed metadata. Ideal systems provide such data to both the host screen and end device.

[0013] Several techniques achieve some subset of these properties. For example, WiFi and Bluetooth are explicit and can be secure, but require configuration and are often not rapid due to configuration overhead. NFC is zero-configuration and rapid, but not inherently targeted or secure. PhoneTouch is explicit, targeted, but not zero-configuration or secure. Other proposed
methods pair devices when they tap one another (using vibro-acoustic and timing data), which is zero-configuration and moderately secure, but not targeted, in that the pairing is only at a device-to-device level.

[00014] Interaction techniques for establishing an ad hoc connection between devices in a convenient but secure fashion have been proposed by several researchers. Such systems differ in terms of the actions users must undertake to establish an association between devices. For example, simple placement of a phone in view of a camera on a specially augmented surface, gestures performed by the user with a mobile device, or synchronous motion or input on two devices, among others.

[00015] The technical means used to establish and verify device associations have been similarly varied. For example, with synchronous device manipulation, matching of inputs has been done with correlation of signals from accelerometers in two devices and, for gesture-based actions, correlation between an accelerometer and a vision-based sensor. Notably, many of these systems require the devices to already be connected somehow to correlate the inputs and verify the pairing action.

[00016] A number of systems have used optical transmission and sensing for pairing devices and several have explored the general problem of transmitting data between devices using color-coding. These include techniques using hue differentials for coding and techniques which use color transitions for coding.

[00017] A phone or other device with knowledge of where it is located in space can be used as an auxiliary input and/or display device to augment interaction with a
larger device. For example they may be used as a magic lens style display to provide physical detail+context displays or, in a large context, as a peephole display, which displays a small part of a much larger surface. In particular, one system uses a hue encoding technique to transmit a phone's x-y position on a display to the phone's camera, enabling spatial tracking of the phone on ordinary displays.

[00018] A few systems come close to achieving the desired properties listed above. A system known as "BlueTable" tracks phones on a surface using an overhead camera, and pairs with them via infrared flashes from the IrDA transceiver. This system was able to transfer data at 2.67 bits per second using binary encoding. Another system, known as "FlashLight" transmits pairing data from an infrared touch table to a phone resting on it. A simple color transition scheme is used to transmit approximately one bit per frame to the phone's camera, achieving a transmission rate of 33 bits per second.

**Brief Summary of the Invention**

[00019] The invention described herein operates with commodity capacitive touch sensors, offers significantly higher data transmission rates, explicitly implements and describes the phone tracking approach, and generalizes the approach beyond tabletops to include consumer touch devices.

[00020] The invention, described herein as "CapCam", includes a technique that provides secure, instant and anonymous connections between two devices. CapCam pairs a "cap" device (a device with a capacitive touchscreen) to a "cam" device (a device
with a camera sensor). For example, typical smartphones and tablets can be paired with each other, and these devices can be paired to even larger touchscreens, such as smart whiteboards and touchscreen monitors.

[00021] CapCam uses the cap device's touchscreen to detect and track the cam device, and displays color-modulated pairing data directly underneath the camera once the camera of the cam device is touching the screen of the cap device. The pairing data is used as configuration data for a bidirectional link, such as an ad hoc WiFi or Bluetooth link. The transmitted pairing data enables these links to be established without requiring user configuration. In this way, CapCam provides a unidirectional communication mechanism from the touchscreen of the cap device to the camera of the cam device, which is used to transmit pairing data that can be used to bootstrap a full bidirectional, high-speed link.

[00022] Users explicitly initiate pairing by placing the cam device on the touchscreen of the cap device. The cap device detects the camera and displays the pairing code, which is securely received by the camera sitting directly on top. Users can target the interaction to particular tasks by placing the camera on specific areas of the cap device's screen.

[00023] The present invention makes attacks difficult in four distinct ways: 1) the handshake requires physical presence; 2) only a single user can use a given pairing code; 3) attackers would require a well-positioned, oblique camera to capture the pairing data; and 4) attackers would need to overcome obfuscating patterns produced by the display.
Thus, CapCam achieves some level of all eight of the desired properties – it provides zero-configuration pairing via automatically transmitted pairing codes; it is rapid, capable of establishing links in under a second; anonymous, in that it requires no hardware identifying information to pair devices; explicitly initiated by user action; secure, in that interception of the pairing code is difficult for attackers; and finally, it enables targeted interactions for simultaneous multiple devices, and in the process, can expose rich metadata about paired devices.

**Brief Description of the Drawings**

[00025] Fig. 1 shows an air hockey game as an example of a context-sensitive placement of the cam device on the cap device's touchscreen to initiate a particular application or action. (A) Players are invited to join the game. (B) When players press their phones to the table, CapCam rapidly and anonymously pairs the phones to the display. When two phones are paired, the game begins. (C) Players use their physical phones to deflect the virtual puck. CapCam tracks the phones and their orientations on the screen. (D) Sounds, vibrations and player-specific information appear on the phone, directed by the game through the paired connection.

[00026] Fig. 2 illustrates raw capacitive images on a large touchscreen display (left) and a Nexus 5 smartphone (mid, right). Images are offset from the object. The metallic camera ring of another Nexus 5 smartphone is visible on the right image.

[00027] Fig. 3 presents a sample packet as encoded by the display. The packet starts with a three-frame green-red-blue header, followed by 21 color frames containing 6
bits each (using 4 color levels per channel). The raw transmitted bits are shown below each frame.

[00028] Fig. 4 shows multiple devices simultaneously tracked on the large touch screen.

[00029] Fig. 5 displays raw bit error rate (% of bits flipped) as a function of color level density.

[00030] Fig. 6 displays percentage of each packet needed for error correction to attain a packet loss rate of 1%, as a function of color level density.

[00031] Fig. 7 shows the effective payload data transmission rate as a function of color level density, after applying the error correction necessary for a target packet loss rate of 1%.

[00032] Fig. 8 illustrates a keyboard example. The large screen provides an on-screen keyboard to paired phones, easing text input.

[00033] Fig. 9 presents an authentication demo. Instead of entering a password on a highly-visible on-screen keyboard, users select an account (left) on their phone. The credentials are automatically sent after pairing the phone (right).

[00034] Fig. 10 shows a data transfer between a phone and a desktop. Left: desktop files can be dropped onto the phone. Right: dragging files to the edge of the phone drops them on the desktop.

[00035] Fig. 11 presents an image gallery example. The user simply presses their phone's camera to an image to download it.
Fig. 12 illustrates a business card exchange. After a meeting, electronic business cards are exchanged. Simply pressing one phone's camera to the other initiates CapCam pairing and downloads the requested card.

Fig. 13 shows a "Context+Focus" application. The large display shows low-resolution "context" imagery. After pairing phones to the display with CapCam (not shown), the phones display high-resolution "focus" imagery corresponding to their position on screen, allowing users to inspect finer detail. Users can individually switch anatomical layers and change zoom levels.

**Detailed Description of the Invention**

In addition to describing the CapCam implementation, several new applications and interactions enabled by the pairing technique are described below. Further, the present invention also compares pairing latency, pairing code bandwidth and bit error rate across three different displays.

The cap device in CapCam can be any capacitive touchscreen device, ranging from large public kiosks to small mobile phones. To demonstrate this range, several different off-the-shelf consumer devices were used for development and testing. Importantly, these devices represent a wide range of sizes and pixel densities. It should be noted that any device having the requisite components could be used.
[00040] Smartphone - A stock Nexus 5 running Android 5.0.1 was used, which features a screen resolution of 1080 x 1920 pixels running at 60 fps and a 5” diagonal screen (445 pixels per inch) with a touchscreen update rate of 120 Hz.

[00041] Laptop - Laptops increasingly feature touchscreens, and so a laptop was chosen as the second display platform. A MacBook Pro was selected, however, this model computer does not contain a touchscreen, preventing any interactive demos from being built. The MacBook has a 2880 x 1800 pixel 15” screen (220 PPI) running at 60 fps.

[00042] Large Interactive Surface - As a demonstration of large CapCam-driven interactions, a Microsoft Perceptive Pixel 55” display was used, with a resolution of 1920 x 1080 pixels (40 PPI) at 60 Hz. The display has a touchscreen update rate of 120 Hz.

[00043] The raw capacitive image produced by the touchscreen controller, as shown in Fig. 2, was used to implement the tracking algorithms, which were designed for finding phones and touches simultaneously.

[00044] To initiate the pairing process, the cam device (typically a smartphone or mobile device) should be placed on the touchscreen surface of the cap device with the camera facing the touchscreen of the cap device. If the cam device features both a front-facing camera and a rear-facing camera, the camera on the side of the device opposite the touchscreen should be used on the cam device. Otherwise, the phone’s touchscreen will not be accessible, preventing certain interactions.
On a small cap device, such as a smartphone, the cam devices are tracked by tracking the metallic ring that often surrounds camera modules, as shown in the rightmost image in Fig. 2. On a large interactive cap device, cam devices are discovered by tracking the rectangular capacitive imprint of the entire device, as shown in the leftmost image in Fig. 2. When a phone is placed on the surface, the phone creates a characteristic disturbance under the entire area of the phone. A connected components algorithm is used to detect blobs in the image. The blob size is used to determine whether the blob represents a phone or a finger touch. For phones, a rectangle is then fit to the blob to extract the approximate bounding box of the phone, as shown in Fig. 4. This fitting is achieved by simple brute force: a bounding box is fit to the blob, rotated by each angle from 0° to 90° in steps of 0.5°, and the angle that produces the smallest bounding box is chosen. The entire phone and touch sensing procedure takes less than 5 ms on a laptop running the system.

Because of the lower resolution of the large interactive display's touch sensor, the camera module cannot reliably be segmented from the rest of the phone. Instead, any contact point inside the phone's area is treated as a potential camera. To accurately discern the actual camera's position, the pairing data are varied in different positions - the variant that the phone transmits for pairing thus reveals the precise camera location.

For the cam device, a Nexus 5 smartphone running Android 5.0.1 was used, however, the scope of the invention is not meant to be limited by this choice, but is intended to include any device having a camera and the other requisite
components (i.e., Bluetooth or WiFi, for example). Android 5 provides low-level control over many parameters of the phone's camera, enabling specialized imaging applications like the present invention. In particular, it allows full-manual control over shutter speed, exposure, focus, and color compensation.

[00048] The displays on the cap devices typically have a refresh rate of 60 Hz, accurate to within fractions of a Hz. Due to the rolling shutter on the camera of the exemplar phone, the camera frame will usually contain part of one display frame and part of the next display frame. The point in the image at which this break happens will move as the camera moves in and out of sync with the display. If the break moves across the image too quickly, the captured frames will skip or lag with respect to the display, which will invalidate the received data. Thus, the phone's shutter speed is configured so that it captures frames as close to 30 Hz as possible. The Nexus 5 can do so to within 0.3 Hz, such that the break moves at most 1/3 of the frame per second. This is sufficiently stable for the application.

[00049] On the camera device, multiple sample points (in the preferred embodiment, 8 sample points) are selected across the height of the camera frame. At each point, the observed color sequence is decoded as a packet. Because it is synchronized closely to the display's frame rate, there will be at least one sample point that is not affected by the rolling shutter break for the duration of the transmission, allowing that sample point to receive the message.

[00050] The data packet may have a fixed length, may contain the packet size in the packet, or may be terminated by a terminating color or sequence. Also, the packet header may be changed to indicate different formats for the data packet.
Pairing data is encoded into sequential frames, where each frame consists of a single color. Thus, the pairing data appears visually as a series of flashing solid colors. To match the frame rate of the camera, the frames are transmitted at 30 Hz (thus, when displayed at 60 Hz, each color frame is simply repeated twice). Each packet of pairing data begins with a three-frame header, consisting of one frame each of solid green, red and blue. This sequence is highly unlikely to occur at random, especially at precisely 30 Hz, and so it neatly delineates packets. Furthermore, using one frame of each color enables the camera to calibrate to the display.

Following the header, the pairing data is encoded into the R,G,B color values of each frame, depending on the number of color levels configured, with the R, G and B values representing three channels. For example, if 4 color levels per channel (e.g. channel values 0, 85, 170, 255) are used, then each channel encodes 2 bits of data, and so each frame encodes 6 bits, as shown in Fig. 3. A 150-bit message can thus be transmitted in 28 frames (three header frames plus 25 data frames), in just under 1 second.

Modern operating systems perform sophisticated color correction to match display characteristics with human perception. These color corrections include gamma corrections, chromatic shifts, white-point adjustments, and brightness/contrast changes. These color correction algorithms are bypassed by specifying that the color data is pre-corrected to the sRGB color space, thus allowing for the display of "raw" RGB values to the display.
Due to the differences between LCD displays, there will still be a color mismatch between the display on the cap device and the camera on the cam device. The three-color packet header is used to calibrate the camera, computing a color calibration matrix which reduces the observed crosstalk between different channels and normalizes the sensitivity of the camera's response to each channel.

After these corrections, there may still be residual noise. For example, some LCD displays exhibit slow switching times, resulting in intermediate values being received by the camera. Because of this, some level of error correction is required to accurately transmit packets.

The encoding scheme, coupled with the ability to control the camera's shutter timing and color correction, allows for the present invention to achieve speeds much higher than the color transition approach of the prior art, despite the need for error correction. For typical displays, the present invention can achieve bit rates up to 6 times faster than prior art methods (150 bits per second vs. 25 bits per second).

Once the cam device receives and decodes the pairing data packet, it initiates a connection to the cap device. In one embodiment of the invention, the connection is made over WiFi, though any wireless protocol can be used.

The pairing data transmitted using the color-encoding as described above will contain any data necessary to establish this connection. For example, it could contain a Bluetooth hardware address and PIN, a WiFi ad-hoc BSSID and WPA2 key, or an IP address and port if the devices are on a shared network (e.g. cellular network or WiFi access point). The pairing data packet also preferably contains a
one-time use passcode, or nonce, which identifies the cam device to the cap device and which can be used as a shared secret to encrypt the communication channel.

[00059] Due to inherent noise and non-uniformity from the camera hardware, color space differences, and LCD imperfections, some level of error correction is needed to ensure reliable data transmission. In one variation of the invention, a BCH error-correcting code is appended to each transmitted packet. The precise code configuration used can be optimized based on the capabilities of the screen. In one embodiment, a 42-bit code appended to an 84-bit message was used, capable of correcting any 6 erroneous bits within the 126-bit packet. These parameters were chosen to give a 99% packet transmission rate on the Perceptive Pixel cap device.

[00060] When contact is broken between the cam device and the cap device, the cap device can immediately detect the loss of capacitive tracking and optionally terminate the connection if no further interaction is desired. This facilitates rapid and effortless creation and destruction of ephemeral pairings between the devices. Moreover, this serves to require physical proximity for pairings to exist, which is not possible with a purely wireless solution. A time delay may be implemented between the time that the cap device detects the loss of capacitive tracking with the cam device and the time when the connection is actually broken, to accommodate movement of the cam device with respect to the cap device for some applications.

[00061] In any pairing technology, where information is sent between two systems, security is a primary consideration. Although CapCam was not designed for
transmission of highly sensitive data, it does exhibit several properties that make it reasonably robust against attacks, compared to, for example, entering a PIN into an ATM.

[00062] A co-located attacker with line of sight could conceivably intercept the data sent from the cap device's display to the cam device. This line of sight is significantly hindered by the fact the visible light handshake is rendered under the device, thus requiring a camera to be at an oblique angle relative to the display with a large depth of field. The data transmission can be further obfuscated with false patterns presented around the camera module.

[00063] Should the attacker be successful in visually intercepting the connection data, they would only gain access to the cap device, not the cam device, and then, only for a brief period of time. Preferably, only one connection is permitted per port/password. Thus, the attacker would have to decode the CapCam packet before the cam device. When the true client cam device does attempt to connect (perhaps a few tens of milliseconds later), the cap device can terminate both connections on grounds of suspicious activity.

[00064] Once a cam device is fully connected, the password (nonce) can be used as a cryptographic key for all network communication. Thus, an attacker observing the network would require the password (transmitted by the cap device) to decrypt the communications, and thus would require line-of-sight or the ability to guess the password (in the prototype implementation, a 32-bit number). For high-security applications, a 128-bit random password could be used.
Evaluation of Exemplary Embodiments

An evaluation was conducted to determine various transmission characteristics of CapCam, for example, the bit error rate, packet drop rate, and effective transmission rates over a range of different transmission encodings and devices.

The evaluation was performed using each of the three devices supported by the prototype of the invention to send data: the Nexus 5 smartphone, the Macbook Pro laptop, and the Perceptive Pixel (PPI) multi-touch display. A Nexus 5 smartphone was used in all three cases as a cam device, to receive the data. The display on the cap device is configured to repeatedly flash a random data packet encoded with a random number of color levels (from 2-16 levels per channel) and a random packet length (from 10-20 display frames per packet), with no error correction applied. In each case, the packet was preceded and followed by the standard G,R,B header.

After the display on the cap device flashes the packet, the cam device sends back the received packet (or reports a failure if the packet was not detected) and the result is compared against the original sent packet to determine the bit error rate.

Each device was tested over a period of eight hours. In total, 113,957 packets were sent (over 32,000 per device), and 64 packets were lost, for a packet loss rate of less than 0.06%. In total, 5,106,848 bits of information were transmitted.

For each device and number of color levels, the total raw bit error rate was computed, as shown in the graph in Fig. 5. The error rate increases as the number of levels increases, due to the decreasing separation between adjacent levels. The
minimum amount of error correction needed to ensure that 99% of packets are corrected (a packet loss rate of 1%) was calculated, for each encoding level. The error correction levels are shown in the graph in Fig. 6 for a packet size of 127 bits. In all cases, a BCH error correction scheme was used. Finally, based on the resulting effective payload sizes, the net throughput of the system was computed, shown in the graph in Fig. 7. The results show that there is a consistent "peak" of throughput around 4-6 color levels, past which the increased raw throughput is outweighed by the increase in error rate.

[00071] As the implementation is designed for a power-of-two number of color levels (for simplicity), 8 levels were chosen for the Macbook and 4 levels for the Nexus 5 and Perceptive Pixel implementations, for bit rates of 150, 135 and 110 bits per second, respectively. These rates are sufficient to send a full setup packet (IP, port, 32-bit passcode) in under one second on all devices. For higher-security applications, 1.5 seconds would be sufficient time to transmit a packet containing a 128-bit key.

[00072] Exemplary Applications

[00073] To illustrate the potential of the approach, seven example applications were built demonstrating different use cases.

[00074] Facilitating Input - A simple application was built that allows the user to enter text on their phone using a larger, more comfortable on-screen keyboard, as shown in Fig. 8. The user simply pairs their cam device (i.e., a smartphone) with
the cap device, causing a position-tracked keyboard to appear near their device.

Multiple keyboards and devices could be paired to the same cap device.

[00075] A demo in which the phone provides input for the cap device was also built. With public kiosks, users often have to enter their authentication credentials on a large touchscreen, where they can be easily observed. In this application, users can press their phone to the display to automatically transmit their credentials rapidly and securely, as shown in Fig. 9.

[00076] File & Information Transfer - Data transfer between a phone and a computer is often challenging. In the desktop demo application, users simply pair their cam device (i.e., a smartphone) with the cap device to enable bidirectional drag-and-drop file transfer. Dragging files off the cam device causes them to be copied to the desktop, as shown in the rightmost image in Fig. 10, and vice-versa, as shown in the leftmost image in Fig. 10. The cap device tracks the location of the phone, allowing its physical location to serve as a drop target. File icons and relative radial position are further streamed to render a live preview of the drop location.

[00077] Alternately, users can select files to download by pressing the phone to the file's icon. In the gallery demo, users can grab a high-res image by simply placing their phone camera to it, as shown in Fig. 11. CapCam transmits a secure, shortened URL for the desired content to the phone, which then downloads and displays the image.

[00078] The cap device need not always be a large display. In the phone-to-phone business card demo, shown in Fig. 12, two users can exchange business cards by pressing their phones against each other. CapCam transmits an appropriate pairing
code (e.g. Bluetooth device ID and PIN) and automatically downloads the selected card file.

[00079] Phone as Accessory - The cam device can also be used as an accessory for larger displays. In the context+focus demo, shown in Fig. 13, the phone is used as a high-resolution focus display for the larger, low-resolution cap device display. The phone's position and orientation is precisely tracked and transmitted by the cap device, enabling seamless navigation of content.

[00080] Games - An air-hockey game has been developed as a demo of the invention. In this demo, two users pair their phones to the cap device to start a match. The phones themselves are used to hit the virtual puck, shown in Fig. 1. Furthermore, the phones can provide stereo audio feedback (e.g. puck hit sounds, goal sirens, and ambient audience sounds), and even individualized haptic feedback, for example, short vibrations when the puck is hit, long vibrations when a goal is scored.

[00081] The present invention includes a technique that allows two devices to establish secure, instant and anonymous connections simply by pressing the camera of one device to the touchscreen of another. By exploiting capacitive sensing, CapCam tracks the precise position of the camera device, which combines with rapid pairing to enable a wide range of interactive functionality. From a technical perspective, it has been shown that the present invention can transfer data from the screen on the cap device to the camera on the cam device at up to 150 bits per second with negligible error, far faster than existing color-based transmission schemes. This is used to transmit data needed to establish a faster and more stable
connection. Finally, a wide range of fun and useful applications has been demonstrated that showcase the capabilities of the invention.
We Claim:

1. A system for pairing devices based on physical contact comprising:

   a first device having a computing capability and a display having a co-located input sensor capable of detecting contact with objects, said first device performing the functions of:

   detecting objects in contact with said input sensor and determining when an object in contact with said input sensor is a second device;

   determining one or more possible locations of a camera module located on the surface of said second device;

   transmitting a data packet by modulating said display of said first device at said one or more possible locations of said camera module.

2. The system of claim 1 wherein said input sensor is a touch screen.

3. The system of claim 2 wherein said modulation of said display of said first device comprises displaying a series of colored frames at the one or more possible locations of a camera module on said second device.

4. The system of claim 3 wherein said modulation contains pairing instructions for establishing a communication link between said second device and said first device.
5. The system of claim 4 wherein said first device further performs the functions of:

- awaiting a connection request from said second device for establishing said communication link using said pairing instructions; and
- establishing said communication link with said second device.

6. The system of claim 1 wherein said one or more possible locations of said camera module is determined by detecting a metallic ring surrounding said camera module.

7. The system of claim 6 wherein said one or more possible locations of said camera module is determined by:

- determining the approximate bounds of said second device on said touch screen;
- fitting a smallest bounding box around said second device; and
- transmitting said series of colored frames at different locations, each varying said pairing instructions; and
- determining said location of said camera module based on the variant of pairing instructions returned by said second device.

8. The system of claim 3 wherein each frame in said series of colored frames is a solid color and further wherein each frame encodes bits of data using the [R,G,B] values defining the color of the frame.
9. The system of claim 8 wherein said data packet is initiated by a header.

10. The system of claim 9 wherein said header consists of three colored frames.

11. The system of claim 10 wherein said header indicates the start of the transmission of said data packet.

12. The system of claim 3 wherein said series of colored frames are displayed at a fixed effective frame rate.

13. The system of claim 1 wherein said pairing instructions consists of data necessary to establish a WiFi or Bluetooth connection between said second device and said first device.

14. The system of claim 4 wherein said pairing instructions further consists of a nonce which can be used as a key to encrypt data transmitted between said first device and said second device.

15. The system of claim 1 wherein said data packet also contains error correcting data.
16. The system of claim 15 where a BCH error correcting code is appended to said data packet.

17. The system of claim 5 wherein said first device terminates said communication link with said second device when said first device detects that said second device is no longer in contact with said input sensor.

18. The system of claim 17 wherein said communication link is terminated a predetermined time after said first device detects that said second device is no longer in contact with said input sensor, if said contact is not re-established.

19. The system of claim 1 wherein said first device initiates an application based on the location of said second device on said input sensor.

20. The system of claim 1 wherein said first device can track movements of said second device on said input sensor and can provide information regarding the movement of said second device to a running application on said first device.

21. The system of claim 3 wherein said series of colored frames is displayed by the screen of said first device at the location of said camera module of said second device, and is surrounded by one or more false patterns of colored frames to obfuscate the series of colored frames to third party observers.
22. The system of claim 5 wherein said communication link is terminated if multiple attempts are made to connect to said first device using said pairing instructions transmitted in said data packet.

23. A system for pairing devices based on physical contact comprising:

   a second device having a computing capability, a wireless communication capability and a camera, said second device performing the functions of:

   receiving a data packet through said camera, said data packet consisting of a series of colored frames, said series of color frames containing pairing instructions for establishing a communication link between said second device and a first device;

   retrieving said pairing instructions from said series of colored frames; and

   establishing said communication link with said first device using said pairing instructions.

24. The system of claim 23 wherein said second device performs the further functions of controlling said camera such that said camera receives unaltered color information.

25. The system of claim 23 wherein said data packet also includes a nonce that is used as a key to encrypt data transmitted and received over said communication link.
### INTERNATIONAL SEARCH REPORT

**International application No.**
PCT/US 17/28484

**Classification of Subject Matter**

<table>
<thead>
<tr>
<th>IPC(8)</th>
<th>CPC</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>H04B 7/00 (2017.01), G08B 21/00 (2017.01)</td>
<td>H04W99/00, H04W8/005, H04W64/00, H04W84/18, H04W76/023,</td>
<td></td>
</tr>
</tbody>
</table>

**According to International Patent Classification (IPC) or to both national classification and IPC**

**Fields Searched**

Minimum documentation searched (classification system followed by classification symbols)

See Search History Document

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

See Search History Document

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

See Search History Document

**Documents Considered to be Relevant**

<table>
<thead>
<tr>
<th>Category*</th>
<th>Citation of document, with indication, where appropriate, of the relevant passages</th>
<th>Relevant to claim No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>US 2009/0176451 A1 (Yang et al.), 09 July 2009 (09.07.2009), entire document, especially Abstract; Para [0005], [0027], [0029], [0033], [0036], [0057]</td>
<td>23</td>
</tr>
<tr>
<td>Y</td>
<td>US 2009/0195402 A1 (Izadi et al.), 06 August 2009 (06.08.2009), entire document, especially Abstract; Para [0025]-[0028], [0057], [0082]-[0083]</td>
<td>1-5, 8-22 and 24-25</td>
</tr>
<tr>
<td>A</td>
<td>US 2015/0256522 A1 (Jeffrey), 10 September 2015 (10.09.2015), entire document, especially Abstract; Para [0042], [0052]-[0054]</td>
<td>1-5, 8-22</td>
</tr>
<tr>
<td>A</td>
<td>US 2011/0195665 A1 (Friedlaender), 11 August 2011 (11.08.2011), entire document, especially Abstract; Para [0051]-[0055], [0060]-[0063]</td>
<td>16</td>
</tr>
</tbody>
</table>

* Special categories of cited documents:
  
  "A" document defining the general state of the art which is not considered to be of particular relevance
  
  "E" earlier application or patent but published on or after the international filing date
  
  "L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
  
  "O" document referring to an oral disclosure, use, exhibition or other means
  
  "P" document published prior to the international filing date but later than the priority date claimed

**Date of the actual completion of the international search**

25 June 2017

**Date of mailing of the international search report**

20 July 2017

**Name and mailing address of the ISA/US**

Mail Stop PCT, Attn: ISA/US, Commissioner for Patents
P.O. Box 1450, Alexandria, Virginia 22313-1450

Facsimile No. 371-273-8300

**Authorized officer:**

Lee W. Young

PCT Helpdesk: 571-272-4300

PCT OSP: 571-272-7774

Form PCT/ISA/210 (second sheet) (January 2015)