Inhibiting COVID-19 transmission

Reducing surface contact with a behavioural nudge device

Andrew Lea, MA(Cantab), FBCS, FRSA
Visiting Research Fellow at the University of Brighton
Head of AI at HelloDone
Director of Computing Scientific

March 19th 2020

One Page Summary

This paper describes a wearable device which may help reduce the spread of COVID-19.

Background

The COVID-19 virus spreads both through airborne infection, after an infected person coughs or sneezes, or through surface transmission. Surface transmission happens when someone touches a surface contaminated with the virus, and then touches their face. Since people touch their faces about 15 times and hour, this is a major source of onward infection.

The idea is to warn people, at least some of the time, when the are about to touch their face. This would reduce the probability of infection. It is also a "behavioural nudge", *training* people to not touch their face.

By reducing the rate of face touching the hope is the probability of infection is reduced, and therefore the transmission of COVID-19. Reducing the rate of spread may "flatten the curve" and help the medical services cope.

The Invention

There are two components:

1. Electronically detectable "tags" are worn near the face on glasses, collars, ear-rings, or necklace. This makes the face electronically detectable.
2. An electronic wrist-band, like a smart-watch or fitness tracker, is worn on each wrist. When it detects that it is in proximity to the wearer's face, it beeps or vibrates, warning the wearer not to touch their face, and training them not to do so.

Implementation and Manufacture

The paper goes into some detail as to how this device could operate and be made. To do this, it describes a "proof of concept" that uses magnets as tags and, for speed, a mobile phone as the sensor, which in real use would be replaced by wrist-bands.

Limitations and Issues

Such a device would not alert the wearer to *all* possible touches. However, even reducing the *number* of touches is directly useful and would also still have a training effect. Equally, the device might raise false alerts, but hopefully not too many.

end of summary

Problem Definition

Surface contact transmission. It occurs when the virus ends up on a surface, to be picked up by a new host. Countermeasures include frequent washing with soap, and not touching the face in general or eyes, nose, or mouth in particular.

According to the Behavioural Insight team attached to the British Cabinet Office, a study found we touch our eyes, nose, or lips 15 times an hour - once every four minutes. If that could be reduced, then the probability of acquiring the virus through touch would be reduced, and therefore the rate of spread of the virus reduced too. Reducing the rate of spread may “flatten the curve” and help the medical services cope. The challenge, then, is to help people not to, and to learn not to, touch their faces.

The Invention

This invention is designed to help people avoid face touching and re-train themselves to not touch their face. It is therefore a form of behavioural change “nudge”.

It works by creating a buzzing sensation or audible beep in the wrist whenever the wearer puts either hand near their face, altering them to abandon their face-touching intention. There are several ways to implement this, one of which is discussed below.

Conceptual Design

The system consists of two elements:-

1. One or more **tags** which can be sensed electronically, are mounted near the wearer’s face, such as on the arms of glasses, on head-bands, in ear-rings, on collars, on necklaces or soft collars designed for the purpose. The aim is to make the face electronically detectable.

2. Two lightweight bracelet or **watch-like sensors,** one worn on each wrist, which can detect those tags - and hence the face - at short range. When either sensor detects one of the tags, it emits a vibration similar to a mobile phone or makes a buzz. The sensor will consist of a detector, a battery power supply, a buzzer or vibration motor, a small micro-controller to control it, various auxiliary components, and an enclosure.

The inverse configuration - sensors and vibrators on the glasses, and tags on the wrists - is also possible. This would probably not work as well, if the connection with which wrist is near the face would be lost. It would, though, require only one set of electronics.

Operation

When the wearer raises their hand near their face, the tag is detected, and the vibrator or buzzer is activated, giving notice to the wearer that they are about to touch their face.

The hope is this will (a) forewarn the user to abandon their face-touching intention and (b) thereby help train them not to do so unintentionally.

Variations

Designs are possible based on different tag and sensor pairs, such as:-

- Magnets and magnetometers, as in the proof of concept discussed later;
- RFID tags and RFID sensors;
- Simpler custom RF (radio frequency) tags, with no identifier payload capability, and sensors.

Designs without tags, might also be feasible:

- optical systems mounted on glasses which detect the in-coming hand;
- orientation and movement based systems, perhaps based on accelerometers and gyros found in smart-watches, might be able to infer the relative position of the arm to the body, through machine learning.

Hybrid variants, incorporating more that one of these approaches, would also be possible. More sophisticated electronic systems could use machine learning to more accurately decide when is best to raise an alarm, avoiding spurious or missed alerts. Alerts could be modulated to provide more information, such as indicating to the wearer how near the hand is likely to be.

Issues

False positives - alerts which occur but should not have - alerts when:

- the wearer does not have a hand near the face. This is countered by having several short range tags, rather than a few longer range ones ones, so that detection is at the best range.
- the wearer intentionally has a hand near the face, perhaps after washing with soap. The wearer can either learn to ignore these alerts, or more sophisticated devices may be able to be more specific in when the alerts are issued, which seems an unnecessary complication.
- an unrelated tag triggers the sensor. A system with tag id's, such as RFID, should mitigate this.

Late positives - alerts which occur after the wearer has touched their face:

- These should be rare. Although not helpful in directly reducing face touching, the alert would still have a training effect.

False negatives - alerts which are not issued when they should be:

- This could easily happen if the system is not sufficiently sensitive, in which case more sensitive detectors are needed or more tags.
- Flat batteries would of course disable the system. Sophisticated variants could issue low battery alerts.

A proportion of false negatives do not negate the training effect.

Effectiveness of this Approach

Assuming this invention works at all, then its effectiveness at reducing virus transmission would depend upon:-

- the reduction in face-touching by wearers near contaminated surfaces. It does not need to eliminate all face touching, but only reduce it, in order to reduce the transmission probability.
- the uptake of this invention

A key element is behavioural: would such a system help people not touch, or learn not to touch, their faces? Or might it introduce a false sense of invulnerability? This invention is primarily a behavioural modifier rather than a direct intervention in the virus transmission. In ordinary circumstances, a study would be carried out to see how well it worked.

Proof of Concept ("POC")

POC Implementation

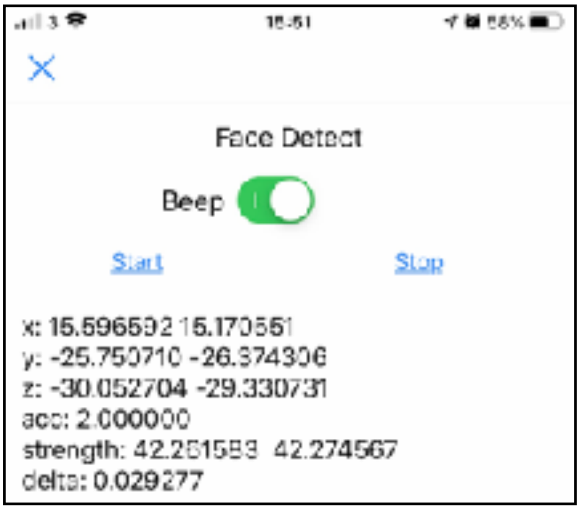
The two elements are:-

1. The tags consist of magnets,

2. The sensors are use magnetometers, which can sense magnetic fields in three dimensions. They are readily available as integrated circuits which can be controlled by micro-controllers, small single-chip computers suitable for use in embedded gadgets. The are also found in mobile phones.

For convenience and speed the POC implementation uses a smartphone, which has been programmed with a small app to respond to changes in the magnetic field. (Details of this are in appendix A.) When the phone gets near the face, it often vibrates and optionally makes a sound.

Testing the POC



The image shows the simple test app display when running. There would be no need for such a display in a deployment.

When the start button is pressed, it starts monitoring the local magnetic field. For the first few seconds are are a flurry of false alarms, which should be ignored.

For testing, the phone was held and the hand moved towards and away from the face. In a deployment there would be no phone, only a smart watch or a wrist band with the detector.

POC Results

The POC was generally able to detect the face on about 50% of occasions. As even 50% of occasions may well be enough for training, the device would have use *even* at this level. However, as there is substantial scope for performance improvement, 80% to 90% might be a realistic expectation.

Pleasingly the false alerts were relatively rare.

Implementation - improving the POC

Hardware

A mobile phone strapped to the wrist is not viable beyond a POC. Two routes seem sensible:-

An app could be developed for **smart watches** and fitness trackers that incorporate magnetometers and preferably accelerometers and gyros.

Bespoke hardware using off-the-shelf components designed to fit in a wrist-band, comprising:-

- a micro-controller, such as an AVR or PIC;
- a magnetometer controlled by the micro-controller using a protocol such as I2C;
- battery, wiring, circuit board, and case

Hobbyists equipped with Arduinos or similar, readily available “breakout” boards with magnetometers, and 3d printers could easily construct these.

Technical Processing Issues

This is a straight forward implementation based on the strength of the magnetic field, in which the <x,y,z> vector was simply converted to a single scaler value.

Magnetometers do not appear suitably sensitive, and even with exponential smoothing, have such noise, that simply using the strength of the field and triggering the alert when it gets strong enough did not yield sufficient range. However, using the rate of change of the smoothed magnetic field strength was - just - sufficient

From an algorithm perspective there are several ways this implementation could be improved:-

- Using “closer to the metal” programming, to receive magnetometer outputs exactly when the device was ready, rather than sampling.
- Optimising the exponential averaging alpha.
- Using other information - such as the orientation of the arm - to preclude certain false alerts would allow the sensitivity to be increased. (You generally can’t touch your face when your arm is pointing down.)
- Combining other device information, such as the device orientation and accelerations from accelerometers and gyros, into a more sophisticated and robust mathematical model.
- Adding buttons for the wearer to indicate false or missing alerts would enable supervised machine learning, so that it could adapt to its wearer.

Caveats

Testing

This POC has only been tested: it was only conceived of 72 hours ago! Ordinarily I would not release a paper like this without building and testing a full prototype. But this is an emergency and other people should also be able to build test evaluate and even manufacture such gadgets.

Peer Review

Although the author has checked with knowledgeable colleagues over the past 24 hours, this paper has *not* been peer-reviewed.

Legal: Risk and IPR

Computing Scientific Ltd issues this paper “as is”. All who agree to assume all the risk, legal or otherwise, of using this invention are hereby licensed to do so; and with that caveat and to that degree I (as a Computing Scientific Ltd director), whilst asserting the moral right to be associated as the author of this paper and invention, in so far as Computing Scientific owns any other IPR rights of in or to this paper and invention, whilst making no representations or warranties that it does own any particular such right, it hereby releases them into the public domain.

Reference

How to stop touching our faces in the wake of the Coronavirus
The Behavioural Insights Team
https://www.bi.team/blogs/how-to-stop-touching-our-faces-in-the-wake-of-the-coronavirus/

Appendix: POC source code

This code was written in the “Pythonista” Python scripting app for the iPhone. It uses a very simply .pyui user interface file with only four essential components, specifically a:

- start button “StartBtn”
- stop button “StopBtn”
- switch “BeepSwitch”
- text view “OutputText”

```
# coding: utf-8

#     Face Detect
#     Andrew Lea, March 2020

#     === Libraries ===

import ui
import motion
import ctypes
import sound
import console

from time import sleep
from math import sqrt

# === Hardware Interface ===

c = ctypes.CDLL(None)

def vibrate():
 p = c.AudioServicesPlaySystemSound
 p.restype, p.argtypes = None, [ctypes.c_int32]
 vibrate_id = 0x00000fff
 p(vibrate_id)

def beep():
      sound.play_effect('digital:Tone1')

#     === Maths ===

def smooth(old, new, k):
      return old*k + new*(1-k)

# === Supervisor ===

Running = False

@ui.in_background
```

```
def StartBtnClick(sender):
      "Monitor magnetic field"
      global Running
      if Running:
            return
      Running = True
      #     == Initialisation ===
      console.set_idle_timer_disabled(True)
      motion.start_updates()
      x,y,z, acc = motion.get_magnetic_field()
      oldStrength = sqrt(x*x + y*y + z*z)
      smoothedStrength = oldStrength
      # == Main Loop ===
      while Running:
            rawX,rawY,rawZ, acc = motion.get_magnetic_field()
            #     Late smoothing regime
            smoothedStrength = smooth(smoothedStrength, sqrt(rawX*rawX +
rawY*rawY + rawZ*rawZ), 0.9)
            #     Early smoothing regime
            x = smooth(x, rawX, 0.9)
            y = smooth(y, rawY, 0.9)
            z = smooth(z, rawZ, 0.9)
            #     Calculate strength of field changes
            strength = sqrt(x*x + y*y + z*z)
            strengthChange = strength - oldStrength
            oldStrength = strength
            if strengthChange > 0.75:
                  vibrate()
                  if v["BeepSwitch"].value:
                        beep()
            #     Display raw and smoothed values for inspection
            v["OutputText"].text = ("x: %f %f\ny: %f %f\nz: %f %f\nacc:
%f\nstrength: %f  %f\ndelta: %f\n" % (rawX, x, rawY, y, rawZ, z, acc,
strength, smoothedStrength, strengthChange ))
            #     Pause to throttle loop rate
            sleep(0.05)
      motion.stop_updates()

def StopBtnClick(sender):
      "Halt main loop"
      global Running
      Running = False
      console.set_idle_timer_disabled(False)

#     === Load interface and run ===

v = ui.load_view()
v.present('sheet')
```