[20230613] INFOMMOB - Mobile interaction - 44 - USP

Course: BETA-INFOMMOB Mobile interaction (INFOMMOB)

Exam questions including some comments on the solutions
(not proofread, thus no guarantee for correctness)

Duration: 2 hours

Number of questions: 9

[20230613] INFOMMOB - Mobile interaction - 44 - USP

Course: Mobile interaction (INFOMMOB)

The exam contains nine questions, each with several sub-questions. Be aware that some questions have lots of sub-questions, others have much less. Also, some may just need a few words to answer, others may need more elaborate text. You can go back and forth between the questions and do not need to answer them sequentially.

You have 120 minutes and can get a maximum of 66 points. Notice that the points do not necessarily reflect the level of difficulty of the sub-question. Therefore, it might well be that a sub-question that gives you the similar number of credits than another one can take longer to answer.

Good luck!

Number of questions: 9

1 General aspects of mobile interaction

In the chapter "*Mobile Computing*" of *The Encyclopedia of Human-Computer Interaction*, J. Kjeldskov describes the history of mobile computing by discussing seven waves or trends.

2 pt. **a.** [max. 2 pts] One of these trends is **divergence**. Shortly describe what that means and give an example of a device commonly used today that could be classified under this trend.

2 pt. **b.** [max. 2 pts] Another trend is **convergence**. Shortly describe what that means and give an example of a device commonly used today that could be classified under this trend.

2 pt. **c.** [max. 2 pts] Two other trends are **connectivity** and **digital ecosystems**. Give an example for mobile interaction that involves at least two devices that form a digital ecosystem (i.e., are connected via a network for short- or long-range communication).

2 pt. **d.** [max. 2 pts] Another trend that made mobile interaction possible was **miniaturization**. Because of that, we now also have high-quality digital cameras in mobile phones. Give one example where the camera of a mobile phone is used for interaction (and not for picture taking).

a+b. See lecture and article. Fun fact: Since it didn't say "digital device," technically, something like "steak knife" and "Swiss army knife" would have been correct examples here, too. Many gave obvious digital examples though.

c. The example that I mentioned in the lecture was "music streamed to the phone, played via external speakers, and controlled via a smartwatch, all connected via a Bluetooth." In the second lecture, I mentioned a weighting scale that sends data to the phone, so it does not need to be entered manually. In the gaming lecture we saw another one (phone used as controller for console game). Plenty other examples exist and were given correctly.

d. See lecture. Plenty examples were given in the first lecture, but we saw several in later ones, too.

Overall, most people answered this question correctly. It was also intended as an easy start into the exam.

2 Sensors for tilt and orientation-based interaction

[max. 2 pts] Give an example for mobile interaction that only requires orientation of a device relative to itself and state what sensor(s) you would use to realize that.

1 pt. **a.** Example:

1 pt. **b.** Sensor(s):

[max. 2 pts] Give an example for mobile interaction that also requires orientation of a device relative to the world and state what sensor(s) you would use to realize that.

1 pt. **c.** Example:

1 pt. **d.** Sensor(s):

[max. 2 pts] Give an example for mobile interaction that requires relative orientation of a device and its absolute location on earth and state what sensor(s) you would use to realize that.

1 pt. **e.** Example:

1 pt. **f.** Sensor(s):

Slide 64 in lecture 2 basically contains the answer to all these questions.
Obviously, various other correct examples and answers exist (e.g., many mentioned "automatically adjust screen to portrait or landscape mode" as an example for the first one, which is correct as well).

3 Touch interaction

In the lectures, we saw some examples for **back of device interaction**, that is, approaches where the back of the device is used for touch input.

1 pt. **a.** [max. 1 pt] Name one common touch interaction problem that is resolved by back of device interaction.

Occluding parts of the screen with your fingers.
(Other correct answers exist.)

The so-called **Midas touch problem** is a potential problem that can appear with both regular and back of device touch interaction.

1 pt. **b.** [max. 1 pt] Explain shortly what it means. (*A few words can be sufficient to answer this correctly. You do not need to explain the Greek mythology that inspired the name for this problem.*)

Various correct ways exist to explain this. One option:

In contrast to other input devices, such as a mouse, which feature a hover mode, touch interaction immediately creates an input when you touch something. This can create a problem when, for example, you do not hit the intended spot correctly and thus create an input at an unintended position.

*A simple answer such as "*no hover mode*" gave full credits, too, since it still reflects that you understand the major issue here. Another short answer that would be sufficient to get full credits: "*Every touch creates an input.*"*

Give one example for regular touch interaction design that deals with this problem. *("Regular touch interaction" refers to common interaction with a touch screen, on your phone, i.e., not back of device interaction. We saw a concrete one in the lectures, but any convincing example is fine and will give full credits.)*

1 pt. **c.** [max. 1 pt] Describe the problem:

1 pt. **d.** [max. 1 pt] Describe the solution:

Various examples exist. One is "callouts" shown on slide 12 in lecture 5.

Problem here: The actual touch location is hidden behind the big finger, which is why we might accidentally hit the screen at the wrong spot. Releasing the finger immediately causes an action, which will then not be the intended one.

Solution: By showing an enlarged version of the covered area above your finger, you can move your finger to the side if you realize you are touching the wrong spot, thus avoiding a wrong input.

Other good examples people used to answer this question: Tilting to hover and touch to click (similar to SWiM, although this is not exactly how they used it there, but many used it as an inspiration to come up with a good answer here). Others referred to the TapBoard, which is a nice example, too.

Some misunderstood what "Describe the problem" was referring to. Since these are solutions that do not resolve all "Midas problems" but only ones in a specific context, this part was intended to describe this context. But if your answer to part d. suggested that you understood the issue well, an incomplete or wrong answer to part c. was not negatively considered.

For back of device interaction, we always have to deal with the Midas touch problem.

1 pt. **e.** [max. 1 pt] Shortly explain why.

Because we don't see our fingers and thus, without a hover mode, we don't know if the spot we are touching is indeed the one we intend to touch. In contrast to interaction on the front, this is always the case.

Illustrate one way on how we could deal with this. *(We saw two examples in the lecture. There might be others. You must only describe one of them. It may help to read both of the following sub-questions before answering the first one.)*

1 pt. **f.** [max. 1 pt] Shortly describe how one can deal with the Midas problem for back of device interaction:

1 pt. **g.** [max. 1 pt] What kind of touch technology is needed for the solution that you described in the previous sub-question?

The table on slide 49 in lecture 3 summarizes the two solutions we saw in the lecture. The preceding slides explain the problem and the solutions both authors came up with. Depending on which solution is implemented, either optical touch technology or pressure-sensitive touch screens are needed.

Other solutions exist. E.g., someone wrote something like: Use the back of device to hover (i.e., place a cursor on the position one wants to click) and tap with your thumb on the screen to create a click. I think that could work, too, and thus got full credits.

4 Multimodality & mobile evaluation

In the paper *Learn with Haptics: Improving Vocabulary Recall with Free-form Digital Annotation on Touchscreen Mobiles,* S. Sheshadri et al. evaluate vibrotactile feedback (VFT) in the context of vocabulary learning.

2 pt. **a.** [max. 2 pts] The authors mention two potential benefits of VFT in the context of vocabulary learning. What are these?

- ⇨ Improves recall / learning.
- ⇨ Feels more natural / better user experience.

These are the two explicitly mentioned and highlighted as major motivation throughout the paper, and which I had in mind when creating this question. Yet, there are others (e.g., implicitly stated in the hypotheses from the pilot studies), which gave full credits, too. "Faster learning and improved recall" was considered a correct answer, too.

To prove their statements, they start with some pilot experiments, followed by an empirical study. For the latter, as for any empirical study, they use independent and dependent variables.

1 pt. **b.** [max. 1 pt] What is an independent variable? Shortly explain. *(Write down the general definition. An example from the paper is asked for in the next sub-question.)*

Independent variables: Variables that we control.

1 pt. **c.** [maxl 1 pt] Name one independent variable used in their experiment.

See top of Fig. 4, which nicely summarizes them.

In empirical studies, we also have to deal with so-called confounding or extraneous variables.

1 pt. **d.** [max. 1 pt] What is a confounding variable? Shortly explain. *(Write down the general definition. An example from the paper is asked for in the next sub-question.)*

Confounding variables: Variables other than the independent variables that might affect the dependent variables.

1 pt. **e.** [max. 1 pt] Name one confounding variable addressed in the paper and how the authors dealt with it.

The issue directly addressed in the paper is the usage of an artificial corpus to avoid that people were presented with words that they were either familiar with because they knew them or because they were similar to words in a language that they knew. E.g. (quotes from paper):

"To avoid any confounds introducing by items which are easier to learn than the rest, cognates had to be avoided."

"We … used the Wuggy pseudoword generator to generate the pseudo corpus" in order to "ascertain that it was participants' first exposure to each of the vocabulary items."

Informal descriptions were sufficient to get full credits. Others exist and got full credits, too. Including ones that were only indirectly mentioned (e.g., counterbalanced order => avoid learning effects). Note that "avoiding gender impact due to gender balancing participants" is debatable here due to the small sample size (12 participants, 5 females isn't really a reliable gender balance). Thus, this answer got only partial credits, because it showed that you understand the issue, but the question was to apply it to a concrete example.

Empirical studies are often discussed with respect to their internal and external validity.

1 pt. **f.** [max. 1 pt] Shortly explain what is meant by internal validity.

Internal validity: To what degree can we assume that the effect does indeed result from the change of the independent variables?

2 pt. **g.** [max. 2 pts] Shortly discuss the internal validity of the empirical study done by the authors (i.e., not the pilot studies). *(Say if it is high or low and provide some evidence for your claim.)*

5 Touch gestures

Given the small screen space of mobile devices, "zooming" is often used in mobile interaction. There are different ways to do this. For example, discrete (e.g., via dedicated "zoom in" and "zoom out" buttons) or continuous (e.g., via "pinch" and "zoom" multitouch gestures).

Give an example where continuous zooming is used and give a convincing reason why this type of zooming makes most sense in this context.

1 pt. **a.** [max. 1 pt] Example:

1 pt. **b.** [max. 1 pt] Reason why using continuous zooming is better here than discrete zoming:

Various examples exist for this and the following question. Thus, here only some general comments.

One criteria of zooming: What zoom levels do you support? Continuous zooming allows you to zoom in to any random level, discrete zooming only to dedicated, pre-defined ones. Which one is better depends on the context. E.g., on a map, we might want to zoom in at random levels, so continuous zooming might be better. For multi-column text, e.g., from a website, zooming in to "parts of a column" doesn't seem to provide a benefit, but it is more intuitive to discretely zoom depending on the column widths.

Another example would be the common problem of "getting lost" or "loosing orientation" in a document due to zooming. Again, depending on the context, this can be easier avoided with continuous zooming or discrete zooming. E.g., for maps, continuous zooming usually helps in maintaining a better understanding of "where you are." For discrete documents, such as thumbnails of a PDF or a photo gallery, discrete zooming with a fixed center might make more sense.

Give an example where discrete zooming is used and give a convincing reason why this type of zooming makes most sense in this context.

1 pt. **c.** [max. 1 pt] Example:

1 pt. **d.** [max. 1 pt] Reason why using discrete zooming is better here than continuous zooming:

There are also situations where zooming cannot help dealing with the problem of small screen estate. Give a convincing example for such a case and shortly explain why.

1 pt. **e.** [max. 1 pt] Example:

1 pt. **f.** [max. 1 pt] Reason why it would not be good to use zooming here:

Again, various examples exist. The one discussed in the lecture was typing on a keyboard. Even if that would make it easier to correctly hit a small key, it is not feasible because we would need to constantly zoom in and out for each key that we type.

In the lectures, we distinguished between three different types of gestures: Ones for *direct manipulation*, *abstract gestures in context*, and *abstract gestures unrelated to content currently shown on the screen.* We also discussed various potential problems that can occur with touch gestures. Name one of these problems and discuss the different types of gestures in this context. *(A short explanation that illustrates if this problem appears here and why is sufficient.)*

1 pt. **g.** [max. 1 pt] A common potential problem with touch gestures is:

1 pt. **h.** [max. 1 pt] Does this problem happen with gestures for *direct manipulation* and why or to what degree?

1 pt. **i.** [max. 1 pt] Does this problem happen with *abstract gestures in context* and why or to what degree?

1 pt. **j.** [max. 1 pt] Does this problem happen with *abstract gestures unrelated to content currently shown on the screen* and why or to what degree?

And again, various examples exist here. One used by many: Gestures and resulting actions need to be memorized. This is easiest for direct manipulation since the gesture matches the intended action. It is harder for abstract gestures made in context since they do not directly map to the resulting action on the screen, but not as hard as for abstract gestures where there is no direct relation between what is seen on the screen and the gesture users must make.

6 Common interaction problems & innovative solutions

In the lectures, we saw three examples that use sensor technology to create innovative solutions to common interaction problems. One of them was the **Force Picker**.

1 pt. **a.** [max. 1 pt] Shortly state what problem(s) the Force Picker approach addresses.

Regular selection menus take up lots of screen estate and need a large gesture space. The Force Picker aims at minimizing them.

1 pt. **b.** [max. 1 pt] What kind of hardware or sensor technology was used to realize this solution?

Force-sensitive touch screen.

Another approach was the **ForceRay.** It uses the same hardware or sensor technology as the Force Picker but addresses a different problem.

1 pt. **c.** [max. 1 pt] Shortly state what problem(s) the ForceRay approach addresses.

Out-of-reach problem / areas on the upper side of the screen are hard to reach when operating a phone with one hand.

1 pt. **d.** [max. 1 pt] Give one potential issue that both approaches might have *(i.e., a general potential interaction problem with solutions that use this type of technology).*

Various correct answers exist. E.g., ergonomics, discoverability, …

1 pt. **e.** [max. 1 pt] Give one potential issue that may occur with the Force Picker *(i.e., a specific potential interaction problem of this concrete approach in this concrete context.)*

Various correct answers exist. Some people did not understand that a "specific potential interaction problem of this concrete approach in this concrete context" was requester here, and not a general problem with force input as above. Hint: make sure to read the question carefully!

Another approach we saw is **SWiM (Shape Writing in Motion)**. It uses a different hardware or sensor technology than the ForceRay but addresses a similar problem (although in a slightly different context, i.e., text entry).

1 pt. **f.** [max. 1 pt] Shortly explain how SWiM resolves this problem.

Instead of directly clicking at a target location, it uses tilt to move a cursor to a position on the screen, and then uses touch (or release of your thumb) to make a selection.

1 pt. **g.** [max. 1 pt] Give one advantage that SWiM might have over the ForceRay approach.

Various correct answers exist. An obvious one (also given by many) would be: It works with most regular phones (since basically all of today's smartphones have an accelerometer) whereas the ForceRay requires dedicated hardware not available in all phones.

7 Mobile gaming

Because the tip of our fingers is quite big, small buttons are often hard to hit accurately. This is why we have guidelines for optimum button sizes for touch screen interaction on mobiles.

1 pt. **a.** [max. 1 pt] Give a convincing reason why this solution (i.e., making the buttons large enough to easily click them) may cause problems in games and motivate game designers to revert to other interaction modes, such as tilting.

Different correct answers exist, including but not limited to:

- Taking up valuable screen estate that could / should be used for game content.
- Using larger buttons will enforce people to move their fingers further, occluding more content and potentially resulting in non-ergonomic interactions.

1 pt. **b.** [max. 1 pt] Give one other aspect or example where touch-based interaction design for games differs significantly from touch interaction design for common contexts or applications such as texting, web browsing, and social media apps.

Different correct answers exist, including but not limited to:

- In games, speed (i.e., how fast you make an input) and accuracy can be even more important (you may lose points or game lives if you miss-click).
- Making the interaction more challenging and harder to master can be part of the fun and thus wanted.
- People prefer (and are used to) rest their fingers on buttons like joysticks.

In the article *A Guide To iOS Twin Stick Shooter Usability,* Graham McAllister explains the difference between a *static* and a *dynamic* implementation of these controls.

1 pt. **c.** [max. 1 pt] Explain the difference between static and dynamic controls.

See paper. Static controls are always visible and at a fixed location. Dynamic ones only appear when the user hits the screen and are placed at this initial clicking location.

1 pt. **d.** [max. 1 pt] Give one convincing example, context, or reason why a game developer might choose for a static control implementation.

Advantages from the slides (but others exist):

- Better control for game designers (e.g., no accidental hiding of info).
- Visualization aid.

The question asked for an example, context, or reason, but just listing an advantage also gave full credits. Ideally, an example or reason was presented associated with this advantage.

1 pt. **e.** [max. 1 pt] Give one convincing example, context, or reason why a game developer might choose for a dynamic control implementation.

Advantages from the slides (but others exist):

- No need for exact targeting.

  The question asked for an example, context, or reason, but just listing an advantage also gave full credits. Ideally, an example or reason was presented associated with this advantage.

Diegesis theory specifies four different interaction design concepts. One of them is a *diegetic representation,* another one is a *non-diegetic representation.*

1 pt. **f.** [max. 1 pt] Shortly explain what a *non-diegetic representation* means.

The interaction component that is visualized is neither part of the game story nor the game space.

1 pt. **g.** [max. 1 pt] Give one example from mobile gaming for a non-diegetic representation and one potential advantage that this implementation might have compared to a diegetic representation of the same example.

Various examples exist. A nice one: In a football game, showing the score and playtime at a fixed location on the screen and in a way that is not part of the game space makes it easier to see by the player. (Compared to integrating it into the game space and story, e.g., via a billboard usually found in stadiums, which would make it appear at different locations and sometimes not be visible at all.)

8 3D interaction

When interacting with 3D content, we often need to change our perspective or point of view of the 3D visualizations shown on the screen. This is often done by moving or rotating the 3D content in the opposite direction than the input. For example, moving an onscreen joystick to the right or tilting the device to the right causes the 3D content or field of view to rotate to the left.

2 pt. **a.** [max. 2 pts] Give an example where it is done like that and shortly explain why.
Various solutions exist, e.g., navigation in 3D worlds.

2 pt. **b.** [max. 2 pts] Give an example where it is not done like that (i.e., where an interaction in one direction does not cause the 3D content or field of view to rotate in the exact opposite direction) and shortly explain why this makes sense in this context.
Various solutions exist. One is mentioned in the question below ☺

These were intended as easy questions since it is mentioned on one of the slides (#24, lecture 8). I was surprised though that several people did not answer them correctly (but many did).

In the lectures, we saw two ways of how perspective projection can be used to create a more realistic 3D effect. One was *Fishtank VR,* the other was *Shoebox VR.*

1 pt. **c.** [max. 1 pt] What sensor(s) are used to implement Fishtank VR?
Accelerometer/gyroscope

1 pt. **d.** [max. 1 pt] What sensors are used to implement Shoebox VR?
User-facing camera

These two were easy, too, and almost everyone got them right ☺

Shoebox VR is generally considered easier to implement than Fishtank VR but has some potential disadvantages compared to the latter.

1 pt. **e.** [max. 1 pt] Give one convincing example where Shoebox VR is sufficient or maybe even the better choice to use than Fishtank VR.
Everything where we can assume that the location of a user's head with respect to the device remains approximately the same (and in a way that makes matches the corrected projection) during the interaction. Maybe the most obvious example is the marble balancing game that we saw in the lecture. Because there it is natural to tilt the device and not move your head.

1 pt. **f.** [max. 1 pt] Give one convincing example where Fishtank VR would likely be the better choice to use or might even be needed.
Everything where we cannot assume that the location of the user's head with respect to the device remain the dame during interaction. An example that we saw in the lecture is the pCubee system, where people are not just tilting the cube but also looking at it from differed angles.

9 Mobile / handheld Augmented Reality (AR)

Touch interactions, such as direct manipulation and touch gestures, work very well for common mobile interaction tasks, but have many disadvantages for AR. **3DMultiTouch** is an approach that addresses one of them.

1 pt. **a.** [max. 1 pt] What is the problem that 3DMultiTouch tries to resolve?

Object manipulation in 3D via a 2D display.

2 pt. **b.** [max. 2 pts] Shortly explain how 3DMultiTouch deals with this.

See video from lecture.

1 pt. **c.** [max. 1 pt] Give one other potential problem or disadvantage of using touch screen interaction for mobile AR.

Some that were listed on a slide in the lecture (but others exist):

- Limited options for interface design (*because, e.g., size is sometimes dictated by real world*)
- Need to hold the device to a certain position (*resulting in uncomfortable holding positions*)
- Need to switch between "touching" real & virtual objects

Thank you for participating in the course. I hope you enjoyed the lectures and look forward to your project presentations.

Once Caracal opens, I would appreciate it if you could give some feedback!