Educational Data Mining toward Personalized Tutoring: Exploration of Facial Behaviors, Thermal Comfort, and Relevant Content Search

Han Jiang

PhD Dissertation in Data Science

Worcester Polytechnic Institute, Worcester, MA

August 5, 2022

Committee Members:
Prof. Jacob Whitehill, Worcester Polytechnic Institute. Adviser.
Prof. Lane Harrison, Worcester Polytechnic Institute.
Prof. Randy Paffenroth, Worcester Polytechnic Institute.
Prof. Andrew Lan, University of Massachusetts Amherst.

Contents

Abstract

We present work on machine learning and educational data mining with the long-term goal of helping to personalize students' learning experiences. The first part is using machine learning methods to analyze videos of students in educational settings. In one project, (1) we explored the relationship between students' thermal comfort, engagement, and learning in a laboratory experiment video dataset and built an end-to-end detector to measure students' thermal comfort and engagement from their faces. In another, (2) we investigated if the empathic messages provided by an intelligent tutoring system could influence the students' emotions and heart rate. In a third, (3) we built a model for predicting when human teachers shift their eye-gaze to look at their students during 1-on-1 math tutoring sessions. The second part is about personalizing educational content by analyzing the detection results of educational videos on YouTube: (4) we compared different methods to provide better math tutorial video recommendations to students by ranking the videos based on the representations conducted by detected math information or the transcripts. Along the way, (5) we found a new kind of training set bias based on the mathematical correctness of object configurations in a visual scene, particularly within the context of detecting individual symbols in images from math tutorial videos.

Chapter 1

Introduction

Machine learning has enabled educational data mining to help teachers better understand students' behaviors and to improve students' learning. Students are different from each other. They may have different preferences. For example, some of them would like to get more attention from their tutors but others would not. They may have different requirements to the tutor and learning environment. For example, some students with strong knowledge background may hope the tutor can spend more time on complex problems while others with poor knowledge background may hope that the basic questions can be explained more times. Thus, personalized learning [123] has became a popular and important direction in the educational data mining area.

Personalized learning in traditional classroom environments is not easy, but there are still some ways how students' learning experience can be improved. For example, schools can provide "smart" desks and chairs in the classroom that can adjust the local temperatures around students. The tutors can allow those students who want more attention, and ask those students who need more attention, to sit closer to the teacher. On the other hand, when students are learning in 1-on-1 tutoring sessions, learning online, or learning with intelligent tutoring systems (ITS), there are more ways in which personalized learning can be implemented to improve students' learning experience.

This dissertation explores several topics within educational data mining towards personalized learning. The first part of our work is focused on analyzing videos of students in educational settings by machine learning methods. These videos contain a lot of information. Students' face and behaviors can reflect the students' learning status, engagement level and emotions [102, 119, 118]. What is a good time for the tutor to get the feedback from the student's face? When the teachers plan to do some actions (e.g., go to next problem, give some hints, give more time), how much should the teacher consider the students' behaviors and how much from the students' facial expressions? Do the hints provided by the teacher have an impact on the students? Understanding the events in the education videos can help to personalize a better learning experience. In one project, we proposed and evaluated a neural network architecture for predicting when human teachers shift their eye-gaze to look at their students during 1-on-1 math tutoring sessions on the SD-MATH dataset [54]. Such models may be useful when developing affect-sensitive intelligent tutoring systems (ITS) [45] because they can function as an attention model that informs the ITS when the student's face, body posture, and other visual cues are most important to observe. Our approach combines both feed-forward (FF) and recurrent (LSTM) components for predicting gaze shifts based on the history of tutoring actions (e.g., request assistance from the teacher, pose a new problem to the student, give a hint, etc.), as well as the teacher's prior gaze events.

With regards to the ITS, most researchers hope that the ITS could be more like a human teacher who can provide supportive, empathetic, or motivational feedback to the learner. But not a lot of research explored whether these feedback messages alter the learner's emotional state or not and whether the ITS could detect the change or not. In another project, we investigated this question on the HBCU dataset [96], which contains 36 African-American undergraduate students who interacted with iPad-based cognitive skills training software [50] that issued various feedback messages. Using both automatic facial expression recognition and heart rate sensors, we estimated the effect of the different messages on short-term changes to students' emotions. Our results indicate that, except for a few specific messages ("Great Job", and "Good Job"), the evidence for the existence of such effects was meager, and the effect sizes were small. Moreover, for the "Good Job" and "Great Job" actions, the effects can easily be explained by the student having recently scored a point, rather than the feedback itself. This suggests that the emotional impact of such feedback, at least in the particular context of our study, is either very small, or it is undetectable by heart rate or facial expression sensors.

Thermal comfort (TC) [23] – how comfortable or satisfied a person is with the temperature of her/his surroundings – is one of the key factors influencing the indoor environmental quality of schools, libraries, and offices. To explore how thermal comfort can impact students' learning and to build a thermal comfort detector, we conducted an experiment in our university. If we can automatically detect the thermal comfort from the students' faces, in the future, some smart building system [20] could personalize the students' local environments to the level that can maximize the students' learning gain. In the experiment, students ($n = 25$) were randomly assigned to different temperature conditions in an office environment ($25°C \rightarrow 30°C$, or $30°C \rightarrow 25°C$) that were implemented using a combination of heaters and air conditioners over a 1.25 hour session. The task of the participants was to learn from tutorial videos on three different topics, and a test was given after each tutorial. The results suggest that (1) changing the room temperature by a few degrees Celsius can stat. sig. impact students' self-reported TC; (2) the relationship between TC and learning exhibited an inverted U-curve, i.e., should be neither too uncomfortable nor too comfortable. We also explored different computer vision and sensor-based approaches to measure students' thermal com- fort automatically. We found that (3) TC can be predicted automatically either from the room temperature or from an infra-red (IR) camera of the face; however, (4) TC prediction from a normal (visible-light) web camera is highly challenging, and only limited predictive power was found in the facial expression features to predict thermal comfort.

Thanks to the development of the internet and existence of large open educational resource repositories, numerous learning resources can be found online. This brings a lot of convenience to the students, but they also need to spend significant time to find a good resource. Providing a better educational content searching [104] service is an important step to achieve personalized searching by the students' preference. In our work, we proposed a new kind of video representation based on the detected math characters that can be used to compare the similarity between the input math expression and the videos. We collected some math tutorial videos from YouTube, sent them to the Amazon Mechanical Turk and asked the workers to watch and label the problems that were solved in the videos. Compared with several different types of representations, the ranking accuracy of our proposed computer vision-based string representation is better than that of the transcripts representation which is our baseline.

When watching a collection of tutorial videos, we found that the quality of the videos was not always good. Some authors provided wrong solutions to the problems. It is widely known that the visual context can affect the object perception in the natural images [41, 3], but can the mathematical correctness influence the character detection results in math tutorial images? To answer this question, we investigate this new type of dataset bias based on the mathematical correctness of object configurations in visual scenes, and how this bias can affect the accuracy of computer vision models. Our experiments demonstrate how CNNs trained to detect and recognize individual objects are capable of implicitly learning simple mathematical relationships between them directly from pixel data; moreover, models that are trained with a dataset bias (e.g., all examples are mathematically correct) can suffer in performance when evaluated on test data without this bias. Importantly, the semantic bias that we study is based not just on simple co-occurrence patterns in each image, but rather on higher-order semantic rules that generalize to unique combinations of objects not seen during training. While the magnitude of the effect was small, the accuracy difference was statistically reliable.

Chapter 2

Predicting when teachers look at their students in 1-on-1 tutoring sessions

The aim of personalized learning in education is to provide personalized feedbacks to students. Student's face can reflect a lot of information, such as students' learning status and their emotions, but their faces not always reflect these information and different students might reflect information at different time. For a teacher, always starring at the student's face will make the students embarrassed. For an ITS, analyzing the students' face in the full tutoring session is too expensive. To achieve the goal of personalized learning in education, to know what is a good time to analyze the student's face or how much we need to consider of the face is very important. In this project, we investigated when the human teacher would look at the student's face from a educational dataset.

2.1 Introduction

Since the early 2000s [64, 65, 28, 121], one of the chief goals of the intelligent tutoring systems (ITS) community has been to develop *affect-sensitive* ITS that can perceive and respond to their students' affective states, e.g., frustration, boredom, and engagement. Due to tremendous, contemporaneous progress in machine learning and computer vision research, the accuracy of automatic detectors of emotions from images and video, both in general (e.g., basic emotions) and educational settings (e.g., detection of student engagement [119, 18]), has increased to the point that they are becoming practical. However, much less research has been done on how automatic affective sensor measurements should be integrated into the ITS' decision-making process.

One key question is: During *which specific moments* of the tutoring session are the students' emotions most important to perceive and respond to? While it is sometimes feasible simply to run an array of detectors on every frame of the videostream (captured from one or even multiple cameras), there are reasons why this is not a good idea: (1) **Computational cost**: as of 2017, the most accurate object detection and recognition systems (e.g., [56, 90, 72]), based on deep convolutional neural networks, are computationally very intensive, more so than "previous generation" detectors such as the classic Viola-Jones [111] approach. In order to maintain real-time responsiveness and low energy cost (particularly relevant for ITS on mobile devices), it may be preferable to sacrifice temporal resolution (i.e., run the detectors less frequently) in exchange for higher recognition accuracy. (2) **Redundancy**: there is a strong correlation between emotion estimates over time. (3) **Data overload**: Estimating a variety of facial expression and emotional states in every video frame can result in a huge amount of data that the ITS must somehow analyze and use to teach more effectively. The magnitude of this data may increase the challenge of training of downstream systems – e.g., a control system that uses "engagement" estimates to adjust the difficulty of the curriculum. It may instead make more sense to attend only to specific moments; indeed, the trend in recent deep-learning research on image- and video-based event recognition is to deploy *neural attention models* [125, 84] that automatically select *dynamically* which parts of an image or video are most salient, based on information contained in the image/video itself. In particular, if the salient moments (when full analysis of all sensors is necessary) can be determined using just a few less computationally expensive, lower-bandwidth sensor readings – e.g., audio rather than video, or low-resolution peripheral vision [43] rather than high-resolution direct gaze – then it is possible that significant computation could be saved.

Human visual attention in one-on-one tutoring: Even in one-on-one tutoring settings, the teacher does not look at her/his pupil during the entire session. In contexts where the student and teacher share a common workspace – e.g., a piece of paper on which to write – the teacher divides her/his attention between the student, workspace, and other objects around the room. The choice of where the teacher decides to look is motivated by several factors, including: (1) **Privacy**: it would likely be uncomfortable for the student to be stared at the entire time; (2) **Information transmission**: From the psychology literature, there is evidence that increased eye gaze by the teacher is associated with more efficient encoding and subsequent recall of information [80, 40, 103] by the student. (3) **Information gathering**: The teacher looks at the student at moments that she/he judges to be most informative for making tutoring decisions. As an example of how these factors can influence visual attention, the teacher might generally avoid looking at the student (to maintain privacy) but decide to "check in" if, after asking her/him to tackle a math problem, the student pauses for a long time without giving any cue that she/he is trying to solve it. This can both help the teacher to know whether the student is confused (information gathering), and it may also cue the student that the teacher is waiting for a response (information transmission).

When developing an ITS that *selectively* perceives its students' emotions, it is necessary to develop an algorithm that decides *when* to look. One approach might be based on reinforcement learning. However, tutoring sessions are relatively expensive and slow to conduct compared to the robotics settings in which reinforcement learning is usually used, likely rendering it impractical. An alternative paradigm, which we pursue in this paper, is to train a model of visual attention using supervised learning from one-on-one tutoring sessions collected from *human* tutors. To the extent that skilled human tutors employ sensible visual attention strategies, this approach could help an affect-sensitive ITS to look at the student during the most important moments.

Human tutors may decide how to shift their eye-gaze based on the high-level actions of the tutoring session – e.g., the student has asked the teacher to help her/him in solving a problem – as well as visual cues such as hand gestures, facial expressions, etc. Tutors' visual attention may also exhibit temporal patterns, e.g., if the teacher just ascertained that the student was "engaged" one second ago, then it might not be necessary to check again during the next second. To date, there has been scant research on how tutors decide when to look at their students (see Related Work); one of the goals of our paper is to start to fill this gap. In one sentence: **the purpose of our work is to explore the extent to which machine learning can be used to predict human tutors' future eye-gaze events, using high-level actions, behavioral cues, as well as the history of prior eye-gaze events, as predictors.**

We emphasize that we are *not* trying to estimate the tutor's *current* eye-gaze (i.e., gaze following [87]) by examining an image of the tutor's face or eye region – this is an interesting and important problem but arguably easier (most human observers can solve this problem easily) than ours. Instead, we are trying to *predict* whether the tutor will *change* her/his eye-gaze during the next time-step. In particular, we assume that the teacher has knowledge of the *high-level actions* (defined in Section 2.3.1) of the session (e.g., give an explanation, request assistance, attempt a problem, etc.); such actions could be obtained, for example, by analyzing the measurements from low-bandwidth (compared to full video) sensors such as speech. We also assume that the teacher knows the history of gaze events she/he has executed so far. Our research harnesses a tutoring video dataset (described below) of two teachers, each of whom tutors 10 middle-school students in a math topic (for a total of 20 unique students), which has been densely annotated for the teachers' (as well as the students') eye-gaze. The focus of our work is on modeling the decision process of human tutors, as well as exploring computational architectures for deciding when to look.

2.2 Related Work

There is a large body of literature [17, 15, 39] on visual saliency and attention prediction. While much of this research focuses on predicting where subjects will look within a single image, there has also been significant prior work on predicting gaze *shifts* in interactive settings, e.g. an airplane flight simulator [29], multi-party conversations [46], and urban driving [16]. To date, there have only been a few studies on visual saliency within *educational* settings: Penaloza, et al. [82] built a model of the *student*'s visual attention to enable a robot to more accurately emulate the cognitive development of infants. We are aware of only 2 prior studies that explicitly model how the *teacher* attends to the student. One is by Dykstra, et al. [32]: on a dataset of 1 teacher with 10 students, they developed a logistic regression-based model that predicts eye gaze shift events (similar to our work) based on the joint actions taken by the tutor and student in one-on-one tutoring sessions. The other is a behavioral study by van den Bogert, et al.[110], who compare expert versus novice teacher's eye-gaze in traditional classrooms (not tutoring sessions).

2.3 SDMATH Dataset

The San Diego Multimodal Adaptive Tutoring Human-to-human (which we call SDMATH) dataset consists of labeled video recordings of 20 one-on-one tutoring sessions. There are 2 tutors in the dataset, one female, one male, both of whom are accredited middle-school math teachers. Each tutor taught 10 students (5 male, 5 female each; no student was taught by both teachers), who were all 8th grade students of 13 years of age. There were 20 unique students in total. Before participating in the tutoring session, both the teachers and the students (and parents) gave informed consent/assent to participate, be videorecorded, and have their face images published in scientific publications (University of California, San Diego's IRB: 090920).

All sessions were captured using both frontal camera to capture student and teacher and an overhead camera to capture the scratch paper which both participants shared as a common workspace (see Fig. 2.4, right). Each tutoring session was approximately one hour in duration and consisted of a 10-minute pretest, 40-minute tutoring session, and finally a 10-minute posttest. The teachers were instructed to teach naturally in order to help each student to practice and learn the material as effectively as possible. The students were instructed simply to do the best they could. The "fundamentals of logarithms" were chosen as the topic of instruction. Logarithms were selected since they were expected to be challenging for the students (since they are typically taught to students in higher grade-levels than the participants in our study) but still learnable to significant degree within a

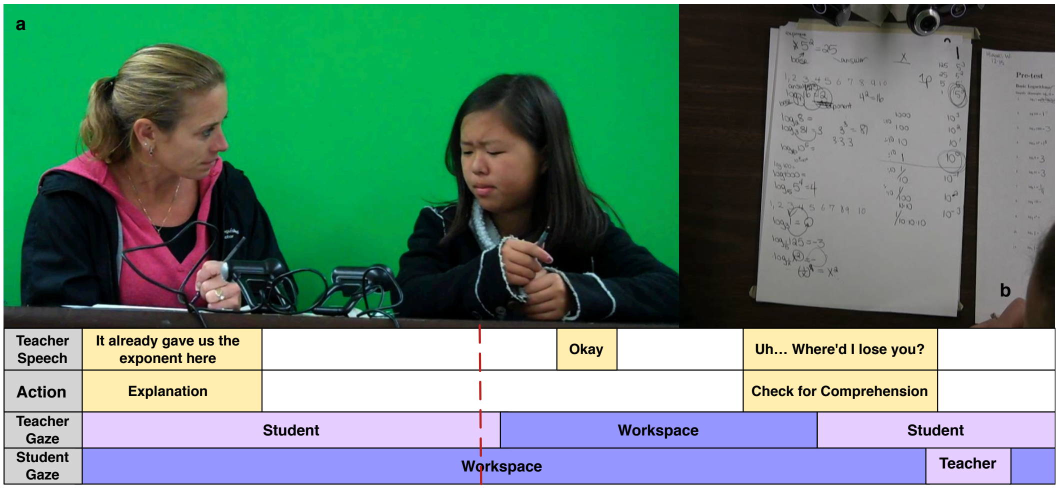


Figure 2.1: A moment from the SDMATH dataset showing both (a) frontal and (b) overhead views. The labels underneath show the teacher's utterances, the corresponding teacher speech action labels, and teacher and student eye gaze labels. The dashed red line indicates the moment at which the image was extracted from the video.

40-minute tutoring session.

2.3.1 Annotation

The SDMATH dataset was annotated for multiple channels (see Figure 2.1 for a schematic): **Actions**: Based both on the teachers' and students' speech, head nods and shakes, as well as the content of what they wrote on the paper, each tutoring session was coded for the *actions* that were taken by each participant at each moment in time. There were 13 possible labels for the teachers' actions (explanation, present problem, solicit content, solicit explanation, solicit procedure, request for participation, provide hint, check for comprehension, direct negation, indirect negation, confirmation, encouragement, and socializing) and 7 for the student (correct attempt, incorrect attempt, incomplete attempt, request assistance, express lack of comprehension, socializing).

Gesture: Hand gestures were coded separately for the left hand and right hand of both the teacher and the student, for all 20 tutoring sessions. Hand gestures were labeled as one of four types (see [75]): *Deictic* (pointing) gestures are used to direct a listener's attention to a referent (e.g., writing on the paper). *Beat* gestures are small hand movements resembling flicks and occur with the rhythm of the speech, mostly placed on stressed syllables. *Iconic* gestures exhibit physical aspects of the scene described by speech. *Metaphoric* gestures are associated with abstract ideas and represent a metaphor of the speaker's idea or feeling about an object or concept.

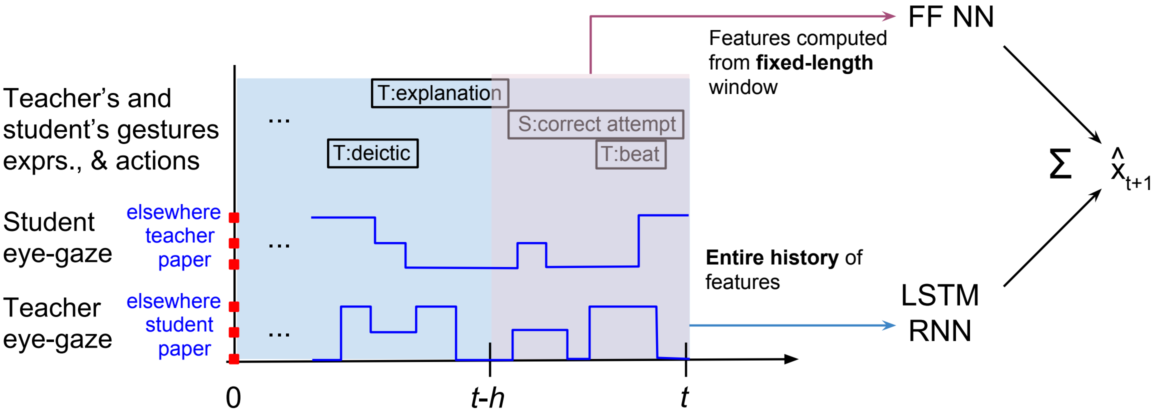


Figure 2.2: Proposed neural network, consisting of both feed-forward (FF) and LSTM components, for predicting whether or not the teacher shifts her/his eye-gaze from {"paper","elsewhere"} to "student", at time $t+1$. The FF network analyzes features computed from a fixed-length window in the pink block; the LSTM analyzes the entire history of the teacher's prior eye-gaze events. The final prediction is the combination of the probabilities of FF NN and LSTM RNN.

Eye Gaze: The object of fixation of student and teacher eye gaze was labeled throughout each tutoring session. Distinctions were made between three mutually exclusive gaze fixations: (1) the paper workspace shared by the teacher and student, (2) the other tutoring session participant (teacher or student depending on the subject of labeling), and (3) elsewhere, defined as all eye gaze which does not fall into one of the first two categories. The median (over all 10 sessions per teacher) fractions of time that the teachers gazed at their students was 6% and 26% for Teachers 1 and 2, respectively.

2.4 Proposed Eye-Gaze Prediction Model

We developed a neural network (see Figure 2.2) to predict the binary outcome of *whether the teacher shifts her/his eye-gaze to look at the student during the next time-step*, based on the history of the student's and teacher's actions (e.g., hand gestures) as well as the prior eye-gaze events of both the student and teacher. In order to capture the *entire* history, we use an LSTM recurrent neural network (see Figure 2.3): the input $[x_t; f_t]$ consists of the *current* eye-gaze x_t at time t, along with the feature vector f_t describing the teacher's and student's actions; the output is the prediction $\hat{x}_{t+1}^{RNN}$ of what the teacher's eye-gaze x_{t+1} (at time $t+1$) will be, over all 3 eye-gaze targets (paper, student, elsewhere).

In addition, since simple feed-forward (FF) neural networks are often easier to train (compared to LSTM) without overfitting, we also use a two-layer FF network to analyze the same set of features (student's and teacher's actions) from the *recent* history over a fixed time-window $[t-h, t]$. The output of the network is a softmax over 2 categories (shift to

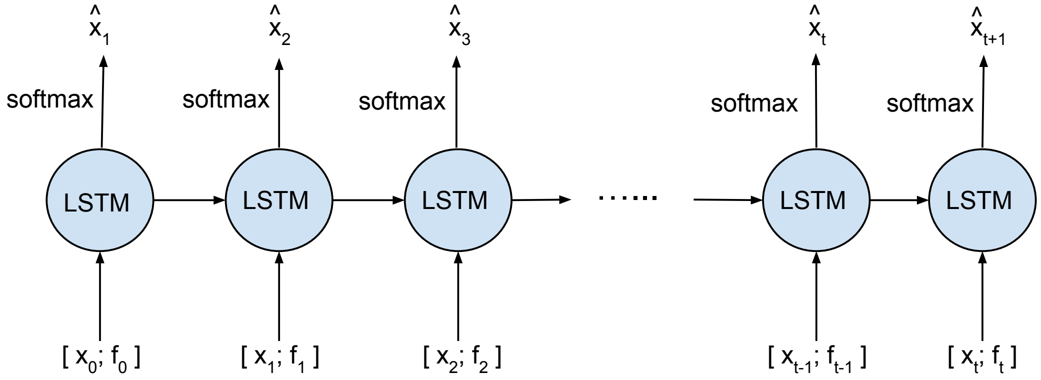


Figure 2.3: LSTM subnetwork we used for eye-gaze prediction. During training, the target value at each timestep t is the ground-truth value of the next timestep $t + 1$.

student, do not shift to student). This is equivalent to logistic regression and is equivalent to the approach used by [32] (though with a different feature set).

The final prediction of the network is the average of the two networks' predictions $(\hat{x}^{FF}_{t+1}, \hat{x}^{RNN}_{t+1})$.

2.4.1 Training

FF: We used as positive examples every time-point at which the teacher shifted her/his eye-gaze from *not* looking at the student (i.e., looking either at the paper or "elsewhere"), to looking at the student. A set of negative examples was created by sampling random timepoints when the teacher was likewise *not* looking at the student and also *did not immediately shift* her/his gaze to the student, subject to the constraint that every such negative example was at least 1 second before the onset and 1 second after the end of every time period during which the teacher gazed at the student. Based on this procedure, there were a total (over all 20 tutoring sessions) of 1836 and 3292 positive examples, and 3652 and 6584 negative examples, for Teacher 1 and Teacher 2, respectively. The value of h was optimized for each teacher to maximize prediction accuracy; this resulted in $h = 0.3$sec for Teacher 1 and $h = 0.2$sec for Teacher 2. The weights of the FF network were also regularized with a ridge term of strength 0.001.

LSTM: In SDMATH, eye-gaze labels are annotated using a *real-valued* clock (e.g., the teacher shifts her/his gaze at time 3.25sec from the paper to "elsewhere"). However, the LSTM recurrent neural network in our design uses a *discrete* clock (each t corresponds to 1 second of wall-clock time). When training the LSTM, we thus set the ground-truth label x_{t+1} that the network is trying to predict at time t to be the proportion of time, within the time interval $[t, t + 1)$, that the teacher gazed at each of the 3 targets. At test time, the outputs $\hat{x}^{RNN}_{t+1}$ were converted (to match the format of $\hat{x}^{FF}_{t+1}$) into a probability vector over just 2 categories by summing the probabilities of "paper" and "elsewhere"; the result was then added to $\hat{x}^{FF}_{t+1}$ to produce the network's final eye-gaze estimate of whether or not the teacher gazes at the student. We trained the LSTM using the Adam optimizer (learning rate was 0.01) over 40 epochs. To optimize the number of hidden units in the LSTM layer (over the set {2,4,8,16,32}), we used subject-independent double cross-validation; the optimal number was 16.

2.5 Results

We used SDMATH to estimate the accuracy of the network described above, for each teacher separately, using leave-one-session-out cross-validation. We measured accuracy separately for the FF and LSTM components, as well as of the overall network (combined predictions). To enable a fair comparison between the FF (real-valued clock) and LSTM (discrete clock) approaches, we tested the network at all timepoints t such that the time interval $[t, t+1)$ contained one of the positive or negative examples used for training+evaluating the FF network. Accuracy was measured using the Area Under the receiver operating characteristics Curve (AUC). Recall that the AUC of a classifier that guesses is 0.5, no matter what the prior class probabilities are.

2.5.1 Results: Predicting *teachers'* eye-gaze shifts

Results (averaged over all 10 students of each teacher) are shown in Table 2.1. The FF network was more accurate than the LSTM network, suggesting that – possibly due to the simplicity of the 2-layer FF network architecture – the short-term history of students' and teachers' actions is more easily capturable using the FF approach than the LSTM approach. However, we did observe evidence that the long-term history of events, as captured by the LSTM, can be helpful: the combined network (FF+LSTM) was statistically significant more accurate (0.79 versus 0.77 AUC for teacher 1, $t(9) = 3.949, p = 0.0036$; 0.70 versus 0.68 AUC for teacher 2, $t(9) = 2.4512, p = 0.03668$) than just the FF network by itself (i.e., the approach used in [32]), suggesting that long temporal windows can be useful for modeling human eye gaze and developing attention models for ITS. Using the combined network, the average AUC over both teachers was 0.75. Clearly, this would not be a high value for an *object recognition* problem such as *gaze following*. However, our problem is about *prediction* and is arguably more challenging.

Teachers' eye-gaze prediction accuracy (AUC)

	FF	LSTM	Combination
Teacher 1	0.77	0.76	**0.79**
Teacher 2	0.68	0.67	**0.70**

Table 2.1: Eye-gaze prediction performance on the SDMATH dataset, using either the FF, LSTM, or combined networks. Results for each teacher are averaged over his/her 10 students.

2.5.2 Results: Predicting *students*' eye-gaze shifts

In addition to modeling *teachers*' eye-gaze, we also "reverse" the prediction problem and train models to predict when the *student* shifts her/his gaze to the teacher. This allows us to train predictive models for not just 2 teachers but also on 20 students, and to gain greater confidence in the ability of our model to generalize to new subjects. Using just the LSTM network (not the FF component, for simplicity), and using the same subject-independent cross-validation scheme (separately for each teacher), we trained predictive models of a student not seen during training. The AUC for predicting students' eye-gaze, averaged over all 10 students of teacher 1, was 0.83; the average AUC over all 10 students of teacher 2 was 0.80. These numbers are consistent with the accuracies of predicting teachers' eye-gaze.

2.6 Identifying the most predictive features

What particular semantic and behavioral features did the teachers in SDMATH respond to when making decisions of where to look? To answer this question, we trained the FF neural network we used sequential additive logistic regression (similar to the FF network described above): For each teacher, we started with an empty feature set and iteratively added the feature (from the pool of 83 features) that maximized the increase in training accuracy, conditional on the already selected features. Selection was repeated for 10 iterations.

Results: The top 10 most predictive features of gaze-to-student events are shown in the tables below, along with the associated logistic regression coefficient:

Teacher 1

#	Person	Feature	Coef.	Cumulative AUC
1	**Teacher**	**deictic gesture (left)**	+.26	0.6231
2	**Teacher**	**explanation**	+.24	0.6745
3	**Teacher**	**prompting**	+.11	0.6917
4	**Teacher**	**check for comprehension**	+.14	0.7113
5	**Teacher**	**beat gesture (left)**	+.13	0.7194
6	Teacher	iconic gesture (left)	+.12	0.7256
7	**Teacher**	**present problem**	−.11	0.7320
8	**Teacher**	**iconic gesture (both)**	+.11	0.7369
9	Teacher	deictic gesture (both)	+.10	0.7430
10	Student	correct attempt	+.07	0.7471

Teacher 2

#	Person	Feature	Coef.	Cumulative AUC
1	**Teacher**	**present problem**	−.25	0.5739
2	**Teacher**	**explanation**	+.13	0.6025
3	**Teacher**	**prompting**	+.17	0.6318
4	Teacher	request for participation	−.08	0.6398
5	**Teacher**	**check for comprehension**	+.12	0.6473
6	**Teacher**	**beat gesture (left)**	+.13	0.6543
7	Student	eye gaze to paper	−.05	0.6600
8	**Teacher**	**deictic gesture (left)**	+.10	0.6646
9	**Teacher**	**iconic gesture (both)**	+.14	0.6694
10	Teacher	beat gesture (both)	+.08	0.6739

Seven out of the 10 features (shown in bold) overlap for the two teachers. The last column shows, for each selected feature, the cumulative accuracy on *training* data. Over both teachers, most of the top 10 features were positively correlated with gaze-to-student, meaning the presence of the feature increased the probability of the teacher shifting his/her gaze to the student. For example, the teachers were more likely to shift their gaze to the student after having started an *explanation*; this is intuitive since the teacher would likely want to sense the student's reaction to what he/she is saying. Similarly, there is a increased probability of gaze-to-student when the teacher *prompts* the student to answer a question, possibly because the teacher is now waiting for the student to deliver a response.

More interesting is that *deictic hand gestures* were positively correlated with the teacher shifting his/her eye gaze to the student. In Figure 2.4, Teacher 2 is shown just before and

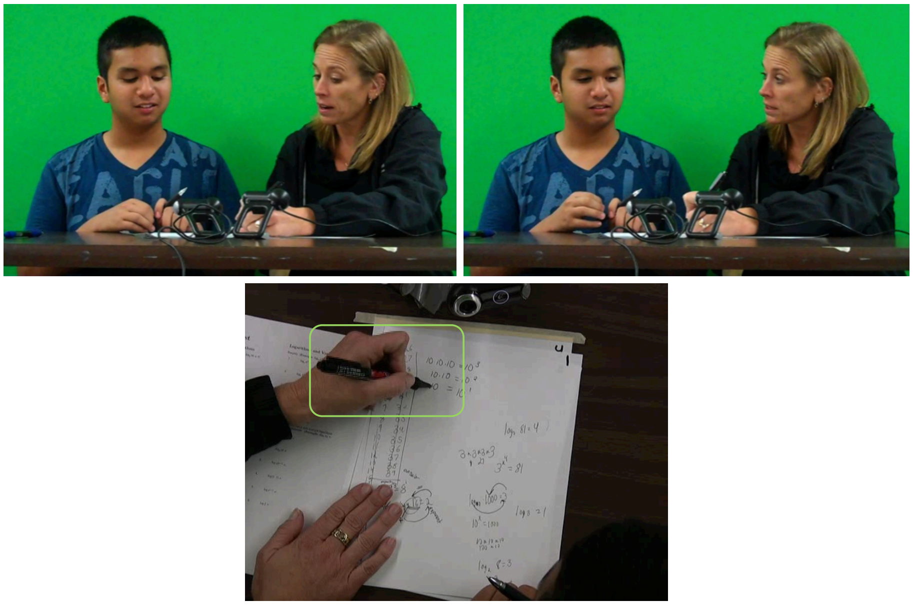


Figure 2.4: Top: Teacher 2 before/after shifting eye gaze to student. **Bottom**: Teacher 2's deictic hand gesture (pointing to an equation on the paper) before shifting eye gaze.

just after she shifts her eye gaze from the paper to the student, along with the overhead view of the paper just before she shifts her gaze. At this moment, the teacher is making a deictic gesture with her left hand to point to the number 10 on the paper. One interpretation is that the teacher needs to gaze at the student to ascertain whether the student is attending to where the teacher had pointed. This suggests that it may be beneficial for an ITS, when pointing out a particular mistake that the student had made in a math derivation, to verify that the student is in fact attending to the tutor's explanation.

2.7 Conclusion

We have proposed a neural network, combining both LSTM and FF components, for predicting whether the teacher in one-on-one tutoring sessions will shift her/his eye gaze to look at the student during the next timestep. This is a challenging problem that requires the model to predict future human behavior. The model was trained and evaluated on a dataset of 20 one-on-one math tutoring sessions from 2 human teachers and exhibited an overall accuracy (averaged over the two teachers) of 0.75 – this corresponds to a reduction in prediction error of about 50% (relative to the baseline guess AUC of 0.5). The accuracy of the overall neural network, comprising both an FF and LSTM component, was statistically significantly higher than just the FF subnetwork, suggesting that long-range temporal dependencies can be useful to capture for predicting eye-gaze events. In addition, we have identified particular high-level semantic actions and behavioral features that the teachers (implicitly) used to make their visual attention decisions. In **future work** it would be interesting to integrate into an affect-sensitive ITS the kind of neural attention model we have developed, and to explore what level of attention prediction accuracy is necessary for the ITS to teach effectively.

Acknowledgement

This material is based upon work supported by the National Science Foundation under Grant No. #1551594.

Chapter 3

Measuring students' thermal comfort and its impact on learning

By automatically detection of the students' thermal comfort, the school buildings could change the local environment to provide a personalized environment to help the students achieve better performance in school. In this project, we explored how to automatically detect thermal comfort and its impact on learning.

3.1 Introduction

Most of the time that people learn takes place indoors. Primary and secondary school students are typically in school buildings for most of the day and do homework in their houses and apartments in the evenings. Adult learners may learn as part of their job in an office or pursue lifelong-learning opportunities at home. The *indoor environment quality* (IEQ) of where people learn, study, and work can have a significant impact on their physical well-being as well as their cognitive performance [4, 7].

The impact of IEQ on *learning* in particular has a special importance and has begun to interest architects, civil engineers, and educational psychologists in recent years [42]: Young learners in particular might be more sensitive to the influence of the environment due to their age or other physiological characteristics than adults. Students spend many hours each day in schools; however, since students typically have little control over their schools' physical environment, learners may feel great concern about their thermal comfort [23]. Thermal comfort (TC), which is a key component of IEQ, has been defined as "that condition of mind that expresses satisfaction with the thermal environment and is assessed by subjective evaluation" [9]. Prior work (see section below) has shown that sub-optimal thermal comfort conditions can negatively affect students' learning. However, to our knowledge, no study to-date has explored the relationship between the impact of TC on learning and *time*. Is it possible that the effect of suboptimal TC could be mild during brief periods of learning but become more severe as the learning session continues? This is one of the questions we explore in this paper.

Measuring thermal comfort: Different people can experience the same temperature and environment differently, and just because one person has a high degree of thermal comfort does not mean her/his friend or peer will. Since thermal comfort is about a person's *satisfaction* with the thermal comfort, it depends not only on the environment itself, but also on the person's physiological and psychological *adaptability* [25, 19] to her/his environment. How adaptive a person is depends, in turn, on how and where a person grew up, e.g., her/his country of origin and its associated climate.

Due to the partially subjective nature of TC, most studies that sought to measure TC used questionnaires [36, 35, 25, 19]. While these are useful, they suffer from drawbacks such as (1) lack of temporal specificity, (2) recency/primacy effects, (3) disruption to regular activities. These can all lead to inaccurate measurements. Therefore, many researchers have explored alternative approaches based on various sensors (e.g., skin-based temperature sensors, cameras) to measure TC automatically [114, 73, 81, 70, 53, 60].

Automatic facial expression recognition: One of the new forms of human observation that has been enabled by advances in machine learning and computer vision is based on automatic facial expression recognition. With technology, it is possible to automatically detect pain in the human body [61], student engagement [119], driver fatigue [113], and many other affective and cognitive states. Inspired by these studies, we explore in this paper whether automatic analysis of facial expression can help to detect a person's degree of thermal comfort.

Contributions: In our study, we (1) conduct a randomized experiment to explore the relationship between thermal comfort, the time-on-task, and learning. We also (2) explore different sensors and algorithmic approaches to estimating a person's thermal comfort automatically.

3.2 Related Work

During the past 10 years there has been substantial interest (see [42, 93] for literature surveys) in measuring the impact of the IEQ on students' learning. In Table **??** we categorize the prior work on this subject in terms of IEQ factor (light, air, etc.) as well as the method of measuring learning (subjective impression (SI), test (T) performance, school scores (SS), and randomized experiment (RE)). In addition to studies specifically about thermal comfort (TC) [129], other factors of the IEQ such as lighting, air quality, and noise have been

Table 3.1: Related Work about the impact of indoor environment factors on learning. SI: subjective impression; T: test; SS: school scores; RE: randomized experiment

	Light	Air	Thermal comfort	Noise	Other
SI	Lee, et al.[67] Choi, et al.[23] Marchand, et al.[74]	Kameda, et al.[62] Lee, et al.[67] Choi, et al.[23]	Lee, et al.[67] Choi, et al.[23] Marchand, et al.[74]	Lee, et al.[67] Choi, et al.[23] Marchand, et al.[74]	
T	Dorizas, et al.[30]	Kameda, et al.[62] Dorizas, et al.[30] Sarbu & Cristian.[94]	Dorizas, et al.[30]	Dorizas, et al.[30]	
SS		Haverinen-Shaughnessy, et al.[48]			Barrett, et al.[13] Barrett, et al.[12]
RE	Marchand, et al.[74]	Wargocki & David.[117]	Wargocki & David.[117] Marchand, et al.[74] Jiang, et al.[57]	Marchand, et al.[74]	

considered. Within this research domain, an important dimension of variability is how learning was measured – by asking participants their subjective impressions, from their school scores, or from a test conducted within the experiment itself. Another dimension of variability is whether the study was observational (i.e., compute a correlation between historical data of the IEQ and historical data of learning) or experimental (i.e., randomly assign participants to conditions). The latter is a generally considered to be the more powerful approach since it avoids many potential confounds (e.g., student engagement) and is the approach we pursue in our study.

3.2.1 Impact of TC on learning

[67, 23] used subjective impression as the learning performance. They both analyzed the relationships between the IEQ (light, air quality, thermal comfort and noise) and learning. [67] found that the learning performance was negatively correlated with the number of student complaints about IEQ. [23] also explored the students' satisfaction with IEQ, as well as the TC in particular, from survey data gathered from 631 university students. The results showed that satisfaction of IEQ of the classroom was related to the perceived effect of IEQ on learning.

[94] conducted a 1-month test during May-June 2012 at a university in Romania. 18 students' test results of concentrated attention tests (Kraeplin test) and distributive attention test (Prague test)[107, 105] were recorded. The conductors used room temperature, relative humidity and CO_2 concentration to predict test scores. Their results suggested that these indoor environment factors could strongly impact students' learning performance. [117] conducted an experiment to explore the impact of air temperature on students' performance. The results indicated that with the same accuracy, students would increase their speed when performing the language-based and numerical performance tasks if the room temperature was reduced from $25°C$ to $20°C$ in late summer. [74] randomly assigned the

participants into different conditions to perform a computer-based reading and learning task. They found that TC had a low and non-significant relationship with the performance; the participants in the extreme condition believed that the temperature had a larger negative impact on their performance than the participants in a normal condition. In [57], the researchers conducted an experiment to explore the impact of TC in 1-on-1 cognitive tasks when students are with a tutor. All the participants experienced all temperature conditions ($10°C$, $14°C$, $15°C$, $16°C$, $18°C$, $20°C$). Their experiment indicated that there was an inverted-U relationship between thermal sensation and pupils' learning performance. A seven point scale of thermal sensation, according to [9], was used. The meaning of the number from -3 to 3 was "cold", "cool", "slightly cool", "neutral", "slightly warm", "warm" and "hot" successfully. The results showed that students' performance was better in the cool or slightly cool conditions compared to the hot condition.

3.2.2 Measuring thermal comfort

How to measure thermal comfort has been explored for many years. While questionnaires from each person about her/his own TC is useful, they can be inconvenient and tedious. Researchers have thus sought to devise alternative measures that can be measured automatically from various sensors.

Environmental sensors: For instance, the PMV-PPD model, proposed by [36, 35], uses air temperature, mean radiant temperature, air velocity, humidity, and human variables to calculate the Predicted Mean Vote (PMV) of a group of people's averaged thermal sensation according to [9]. The Predicted Percentage of Dissatisfied (PPD) utilizes PMV to calculate the percentage of people who might complain about their thermal environment.

Body sensors: [114] used skin temperature sensors to collect upper extremity (finger, hand, forearm) skin temperatures and explored how these temperatures related to thermal sensation. [73] explored different configurations of where to place the temperature sensors on the body and identified particular configurations that were most effective.

Cameras: More recently, with the development of machine vision, researchers also explored predicting thermal comfort through cameras. [81] showed that the averaged forehead temperature from infrared (IR) images was correlated with people's thermal sensation and thermal comfort. [53, 60] leveraged the human thermoregulation process and then applied Eulerian Video Magnification algorithm[122] to filter the visible-light RGB images to predict thermoregulation states, which is one indicator of thermal comfort.

3.3 Experiment

In order to assess the impact of thermal comfort on learning and how this effect could change over time, we conducted a laboratory-based learning experiment (approved by WPI's IRB #18-0372) in which university students ($n = 25$) watched three lecture videos, answered surveys on their thermal comfort, and completed a quiz on what they learned. During the experiment, the indoor environment conditions were monitored and controlled according to a schedule defined by each participant's randomly assigned experimental condition. We also deployed a variety of sensors – camera, environmental, and body – to measure the temperature of the environment and of each participant. These sensor measurements, along with participants' survey responses, allow us also to explore different automated approaches to estimating a person's thermal comfort.

3.3.1 Recruitment of participants

We recruited participants for the experiment through an email list at our university. In the end, 25 students (of whom 9 were female) participated in our experiment. All of them were either undergraduate or graduate students. Each participant was paid for $20 gift card for his/her participation.

3.3.2 Procedure

This experiment was conducted on each participant individually and was divided into four sessions. Each session was 21 minutes. Therefore, every participant would sit at a desk around 84 minutes in total. In the first session (adaptation session), each participant gave informed consent, placed the skin-based temperature sensors on her/his body, and listened to the experimenter's instructions. The purpose of the adaptation session was to neutralize the potential impact of the outside weather conditions or physical activity (e.g., running to class) before the experiment. In each of the remaining three sessions, the participant watched a tutorial video (10 minutes), answered a quiz about it (<5 minutes), completed a thermal comfort survey (<5 minutes), and then took a break. The length of the break (21 min $-$ VideoLength $-$ QuizTime $-$ SurveyTime) depended on how long the participant took to complete the quiz and survey. The order of the tutorial videos was randomized, as was the order of the temperature conditions (warm to neutral, or neutral to warm); see Conditions subsection below. Sensor measurements, including video of the face, were recorded throughout all three tutorial sessions.

After the participant finished putting on the body sensors, the experimenter started the videorecording from the laptop-based web camera, typed the participant's ID into the

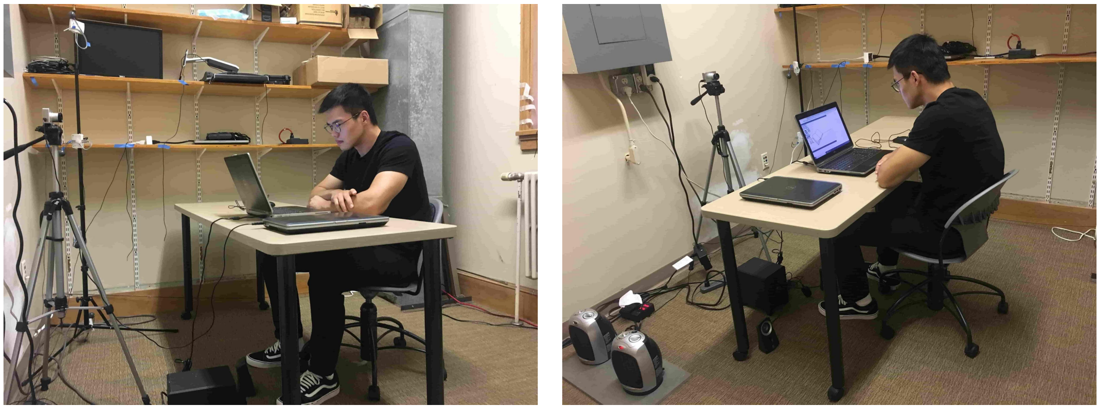


Figure 3.1: Experimental setup of the desk, laptop, and cameras.

web-page, turned the time controller on, and then asked the participant to press the "Start" button whenever she/he was ready. The experimenter then left the room and stayed in the room next-door throughout the rest of the experiment. Using remote access software, the experimenter took an IR image of the participant at the beginning of each tutorial video during the tutorial sessions. See Figure 3.2 for a schematic of the procedure.

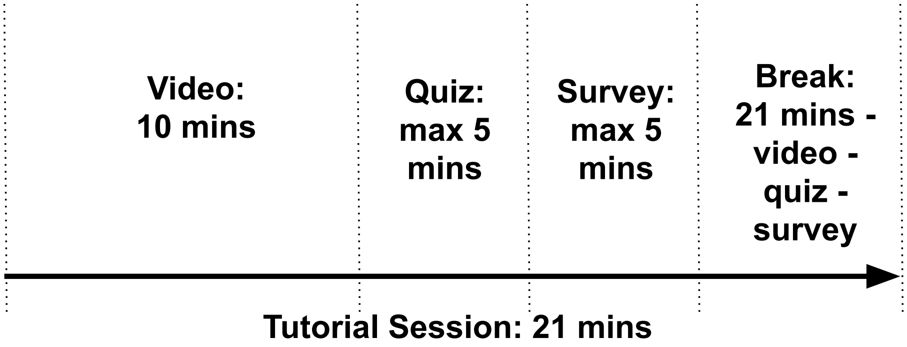

Video: 10 mins
Quiz: max 5 mins
Survey: max 5 mins
Break: 21 mins - video - quiz - survey
Tutorial Session: 21 mins

Figure 3.2: Tutorial session procedure

3.3.3 Environmental controls

We used 4 heaters (to increase the room temperature) and 1 air conditioner (to decrease temperature). In order to maintain the temperature at a constant level, we also deployed 3 thermal controllers. Moreover, in order to change the room temperature (from either warm to neutral, or neutral to warm), we also used 4 timers. To maintain the room temperature to be at least $25°C$, 1 heater was always turned on. 3 thermal controllers and 3 timers were connected to the other heaters. The thermal controllers were used to keep the room temperature around $30°C$. Timers were used to control when the heaters and air conditioners were turned on and off. The heaters and air conditioner were oriented so that the air did not blow directly onto the participant.

3.3.4 Sensors

All sensors were adjusted carefully before we started our experiment. They are listed as follows:

1. 4 skin temperature sensors. We followed the positions in [73] (see Figure 3.3). These sensors were used to measure the participant's body temperatures at four different body locations and record the temperature every 1 minute. Sensors were attached using medical tape.

2. Room temperature sensors. These sensors were used to measure the room air temperature at different heights (0.1m, 0.6m, 1.1m and 1.7m) and recorded every 1 minute.

3. 1 web camera on the laptop pointed at the participant's face. Note that the video was lost for 1 out of 25 participants; hence, for our experiments on using the web camera to predict thermal comfort, $n = 24$.

4. 1 infrared (IR) camera pointed at the participant's face. The camera recorded only images, not video. Using the camera's temperature calibration software, the IR images can be used to estimate the participant's face temperature directly.

3.3.5 Materials

Tutorial videos: We used three 10 min-long tutorial videos and quizzes that were used in a prior study by [109]. The order in which the tutorial videos were presented to each participant was randomized; this was necessary to remove the potential confound that the subject matter, rather than the thermal comfort or time during the learning session, influenced the learning gains. All videos were about social, philosophical, and ethical issues: (1) honesty, (2) language and thought, and (3) empathy.

Thermal comfort survey: We used the same thermal comfort questionnaire survey as in [70, 69]. The survey asks questions such as, "Rate your whole body thermal sensation", "Rate your thermal body comfort", "How sleep/alert do you feel?", and "How easy/difficult is it to concentrate?" The scale was from -3 to +3 with a resolution of 0.1.

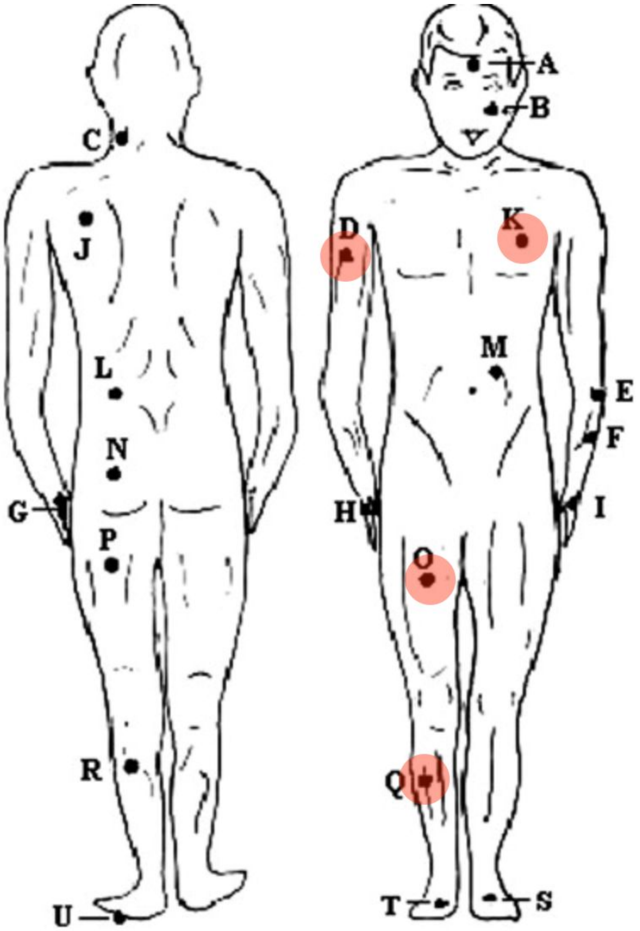


Figure 3.3: Positions of skin-based temperature sensors on the body.

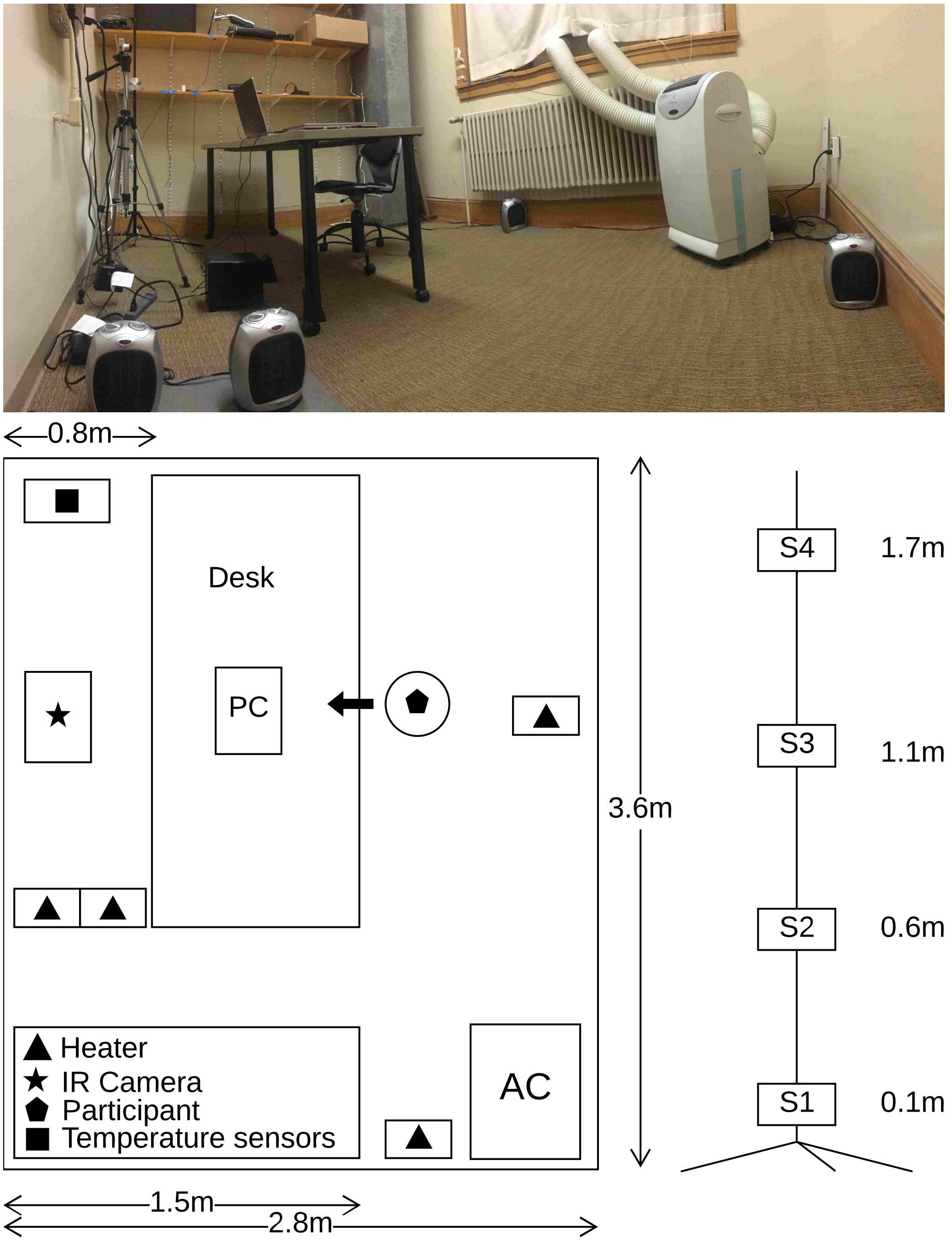


Figure 3.4: Top: Experiment lab Photo; Bottom Left: Top view of Lab and sensors' position. The participant was facing the direction with the arrow; Bottom Right: Room temperature sensors in different heights

3.3.6 Conditions

Each participant was randomly assigned to one of two temperature conditions: neutral to warm (25°C to 30°C), and warm to neutral (30°C to 25°C). By randomizing the thermal

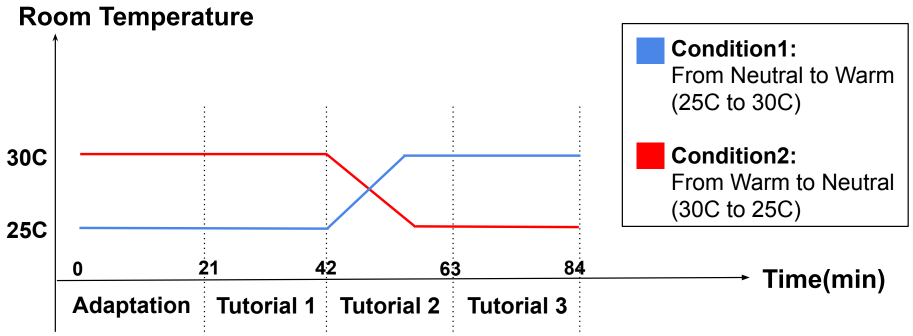


Figure 3.5: The change of room temperature in different conditions

conditions, we avoid the potential confound that students' performance changed in different sessions not due to thermal comfort but due to other factors related to time, e.g., fatigue. If the participant was in the neutral to warm condition, the room temperature in the adaptation session was maintained at 25°C until the end of the first tutorial session; it was then increased to 30°C in the second tutorial session and was maintained at this level until the end of the third tutorial session. See Figure 3.5.

3.3.7 Data collection

Using the sensors, we collected several kinds of data from each person: (1) Video from the web-camera (at 30 fps); (2) Infrared images (1 every 21 minutes); (3) room temperature, CO_2, and relative humidity (1 measurement every minute); (4) body temperature (1 every minute for each sensor); (5) each participant's start/end times of each tutorial video, quiz, and survey; (6) each participant's quiz scores.

3.4 Analysis

Our analysis was focused on two questions: (1) what is the relationship between thermal comfort, temperature, and learning? (2) How can we use the various sensors to estimate participants' self-reported thermal comfort automatically?

3.4.1 Impact of room temperature on thermal comfort

In our experiment, the range of the room temperature was from 25°C to 30°C. This was not a huge change in the temperature. One of our goals was to assess whether this magnitude of temperature change could influence body thermal comfort. As defined in the thermal comfort survey that we used [9], the range of thermal comfort was from -3 to 3, where -3

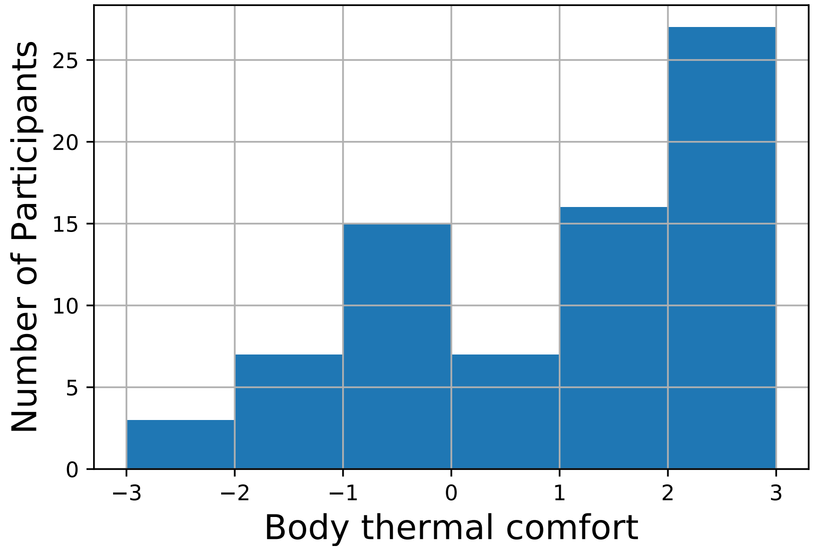


Figure 3.6: Histogram of thermal comfort in our experiment

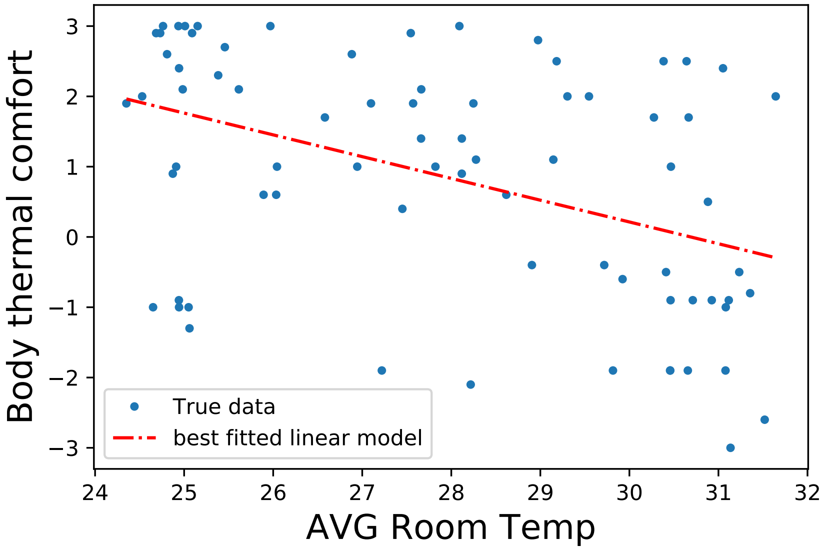


Figure 3.7: Thermal comfort VS Avg room temperature

means “very uncomfortable” and 3 means “very comfortable”. Based on the histogram of body thermal comfort in our experiment in Figure 3.6, we see that the participants rarely (10 total votes) considered their thermal comfort to be highly uncomfortable (a rating of -3, -2). This indicated that our setting of the experiment was relatively comfortable for most of the participants. Did the modest temperature changes induced during the experiment impact participants’ thermal comfort? To investigate, we considered models including either linear or quadratic terms for room temperature (computed as the average of the temperature sensors at different heights). The quadratic model did not give a stat. sig. better model fit, and hence we used a linear model; see Figure 3.7. The Pearson correlation between the model’s predictions and self-reported thermal comfort scores was $r = -0.436$, $p < 0.001$, i.e., within the temperature range of our experiment, higher temperature resulted in lower thermal comfort. Based on the estimated regression coefficient, increasing the room temperature by one degree in our temperature range results in a reduction of thermal comfort by 0.32. Note that we also tried modeling thermal comfort and temperature (linearly) with a participant-specific offset as a random effect and obtained similar results.

3.4.2 Relationship between thermal comfort, learning, and time

After showing the change of room temperature in our experiment could influence the participants’ thermal comfort, we assessed whether thermal comfort was related to participants’ performance in the learning task. A scatter-plot of the quiz scores versus self-reported thermal comfort scores is shown in Figure 3.8. Neither the Pearson nor the Spearman correlations between quiz score and thermal comfort were significant. However, after visually examining the scatter-plot, we noticed a slight ”inverted U” shape; this has also been noted in prior work [100, 99]. This shape indicates that when the participants felt too comfortable or too uncomfortable, their quiz score were lower; when the thermal comfort state was in the middle, their quiz score was higher. We found some support for this hypothesis in our data: the Spearman correlation between the *square* of self-reported thermal comfort and quiz score was negative ($r = -0.235$) and statistically significant ($p = 0.0042$). Tthe quadratic model of self-reported thermal comfort gives a stat. sig. better fit than the linear model (likelihood ratio test, $p = 0.002$).

To explore this more rigorously by accounting for repeated measures, we also used a mixed-effect model with a random effect to model an offset for each unique participant. Due to different tutorial videos having different difficulties, we also considered the video_id as the random effect. We studied the relationship between thermal comfort and quiz score within each of the three tutorial session (1, 2, 3) separately. To our surprise, in the first two tutorial session, the impact of the square of the body thermal comfort (i.e.,

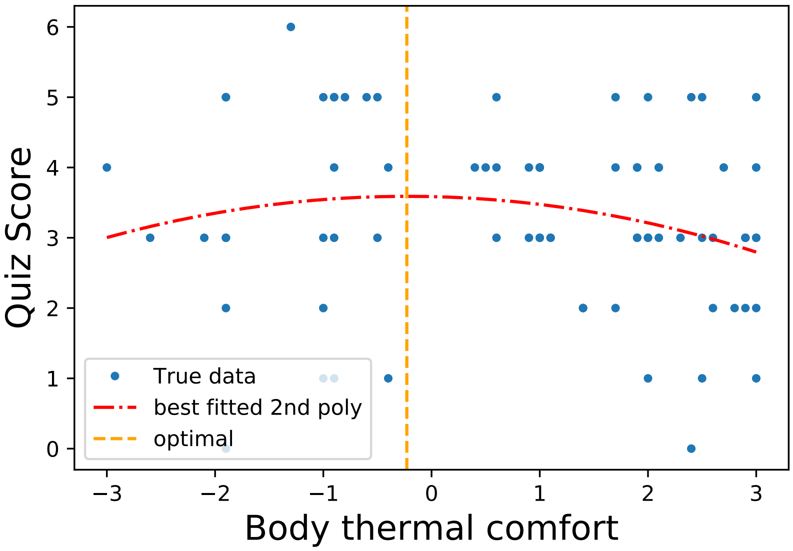


Figure 3.8: Thermal comfort VS Quiz score

Table 3.2: Effect size (Cohen's f^2) of TC^2 in each tutorial session

Session No.	Effect size
1	0.007
2	0.044
3	0.308

TC^2) was not significant ($p > 0.05$). However, in the last (third) session, the impact was negative and stat. sig. ($p = 0.013$). The estimated magnitude was that a change in 1 level of thermal comfort decreases the quiz score by 0.2 points (the maximum score was 6 points). A possible interpretation is that, as time went on, the participants might feel more tired or bored. At first, they could force themselves to focus on the tutorial videos and answer questions. However, when they became fatigued or bored, an uncomfortable thermal comfort might start to show its influence. See Table 3.2 for the effect size(calculated based on the marginal R^2) in each tutorial session.

3.4.3 Relationship between thermal comfort and sleepiness

The survey that each participant completed after every tutorial session contained questions not just about thermal comfort, but also about how sleepy they felt. The values ranged from -3 (very sleepy) to +3 (very alert). The correlation between thermal comfort and sleepiness was positive (0.32) and stat. sig. ($p = 0.0084$).

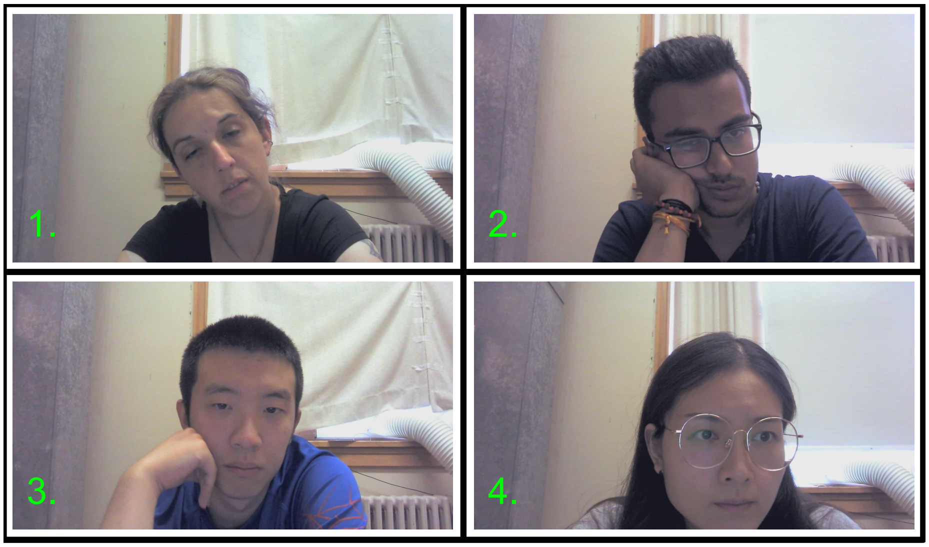


Figure 3.9: Participants in different engagement levels.

3.4.4 Relationship between engagement and learning

To explore whether the perceived level of student engagement, as judged by an external observer, was related to students' learning, we manually labeled video frames from each participant's face video. We extracted 1 frame every 20 seconds for each of the 3 tutorial sessions of all the participants. These pictures were labeled for the appearance of 'engagement' following the definitions in [119]. Level 1 is "not engaged", level 2 is "nominally engaged", level 3 is "engaged", and level 4 is "very engaged"; see Figure **??** for a representative image of each label. During labeling, the images were randomized over time and also over participants; hence, the engagement scores were unbiased w.r.t. participants' self-reported thermal comfort. We averaged the engagement for each participant per each of the three tutorial sessions, and then used a mixed effect model to analyze the relationship between quiz score and engagement. The participant_id was still the random effect. Since we had a prior hypothesis that engagement was positively correlated with learning, we used a 1-tailed t-test. The result showed that this positive correlation was significant ($p = 0.032$).

3.5 Automatic detection of thermal comfort

The primary method of estimating thermal comfort is via self-report on a survey. Might there be an automated way of obtaining this information that is less intrusive and gives higher temporal resolution? This could be useful to advance research on the IEQ and

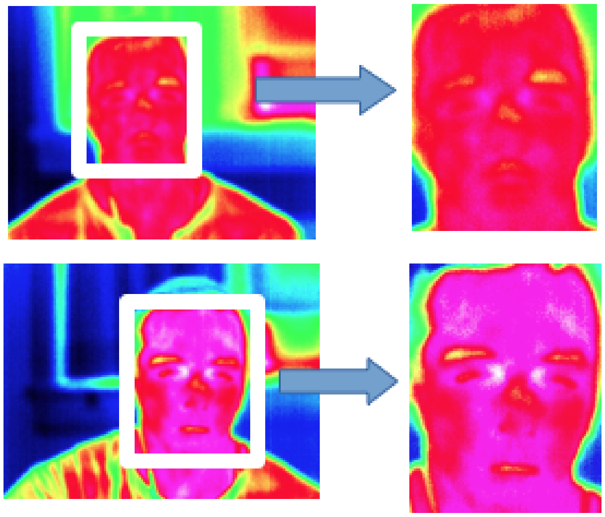


Figure 3.10: Manually cropped face for infrared images. Top: face when thermal comfort is -0.6. Bottom: face when thermal comfort is 2.7.

learning. Moreover, it could also set the stage for smart learning environments in which localized ventilation, heating, and cooling systems can optimize the thermal comfort for each learner. With these goals in mind, we explored several approaches to automatically estimating thermal comfort using the different sensors we deployed in our experiment.

3.5.1 Infrared camera

Per participant, 3 IR images were collected (one per tutorial session). From each IR image, we manually cropped the face for infrared images from IR camera and calculated the average face temperature for each tutorial session. For each IR image, we cropped the face between two ears for width, and from forehead to chin for length; see Figure 3.10.

We then calculated the mean temperature within the face region and used it to predict thermal comfort. Using a mixed-effect model (with participant_id as a random effect), we found that the correlation between the face temperature, as computed from the calibrated IR image, and thermal comfort was $-.34$ ($p = 0.0029$). In other words, a hotter face was associated with lower thermal comfort.

3.5.2 Skin sensors of body temperature

We averaged the skin temperature from 4 skin sensors for each tutorial session. The correlations between thermal comfort and averaged skin temperature are shown in Table 3.3.

With statistical significance, the correlations of the skin temperature at position D and

Table 3.3: Skin Temp. VS Thermal comfort

Sensor	Pearson Correlation	p-value
D	**-0.273**	**0.018**
K	-0.174	0.136
O	-0.186	0.11
Q	**-0.28**	**0.015**

Q indicated that they had a negative correlation with body thermal comfort. These two correlations also remained significant when we applied the mixed-effect model and set participant_id as random effect.

3.5.3 Web camera

Even though the results of skin sensors and infrared cameras showed that we could use them to detect thermal comfort, we were still interested in whether an ordinary (visible light) web camera can be used to detect thermal comfort. In contrast to skin sensors, web cameras are less intrusive – they require no skin contact or medical tape. In contrast to IR cameras, they are less expensive and more widely available.

While one could consider a "black box" approach such as a CNN-LSTM in which all the pixels of an entire video segment is used to predict thermal comfort, the relatively small size of our dataset ($n = 24$) makes this approach difficult. Instead, we investigated whether the much lower-dimensional feature representation of facial expressions can reveal a person's thermal comfort. For example, we reported above that sleepiness is associated with thermal comfort, and this might be revealed in a person's facial expression; this approach was used in [113] to detect drowsiness when driving a car.

After watching the videos, our subjective impression was that predicting thermal comfort from the face was very difficult. In the temperature range of our experiment setting, the facial expressions in different temperature condition did not vary greatly. Nevertheless, we tried three approaches: (1) estimate thermal comfort directly from the average facial features values extracted from OpenFace [11] over the time series of face images; (2) estimate thermal comfort from a Gabor-filtered time series of facial features; and (3) train a recurrent neural network to analyze the raw time series.

Individual face movements

From each frame in each 10-minute video sequence just prior to the self-reported thermal comfort survey of each tutorial session of each participant, we used OpenFace to extract the facial action units (AUs 1, 2, 4, 5, 6, 7, 9, 10, 12, 14, 15, 17, 20, 23, 25, 26, 45). In

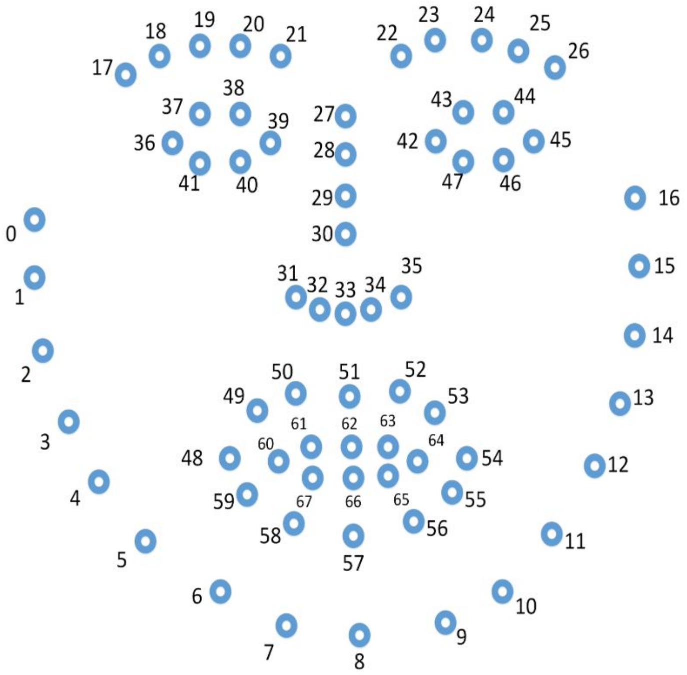


Figure 3.11: Landmarks from OpenFace

addition, we also calculated the size of the face – this could be useful for determining if the participant leaned toward or away from the camera. Next, we extracted the head pose. Finally, we computed the distance between the eye-lids – this could give some measure of drowsiness.

For the left eye, we first calculated the central point of landmark 37 and 38, the central point of landmark 41 and 40, and then, calculated the distance between the these two central points. For the right eye, we calculated the distance used landmark 43, 44, 47 and 46 as the same approach as the left eye. The eye-lid distance was the mean of the left distance and the right distance. We also estimated the size of the face box as an indication of whether a person was leaning towards or away from the camera: we first calculated the central of landmark 19 and 24, and then calculated the distance between the central and landmark 8, and also the distance between the landmark 0 and 16. The final face size was the product of the two distance. See Figure 3.11.

Using the above feature set, we examined the Pearson correlation between each mean feature value (averaged over each 10-minute time series) and self-reported thermal comfort. Only two features were stat. sig. correlated: AU 6 (Pearson $r = 0.244$, $p = 0.038$; see Figure 3.12) – cheek raiser – and the eye-lid distance, calculated by the landmarks on the eyes, was also correlated to thermal comfort with significant (Spearman $r = -0.27$, $p = 0.02$). The latter correlation suggests that smaller eye opening is associated with larger

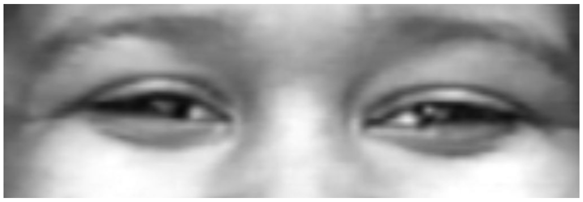


Figure 3.12: Example of AU 6 (`https://www.cs.cmu.edu/~face/facs.htm`

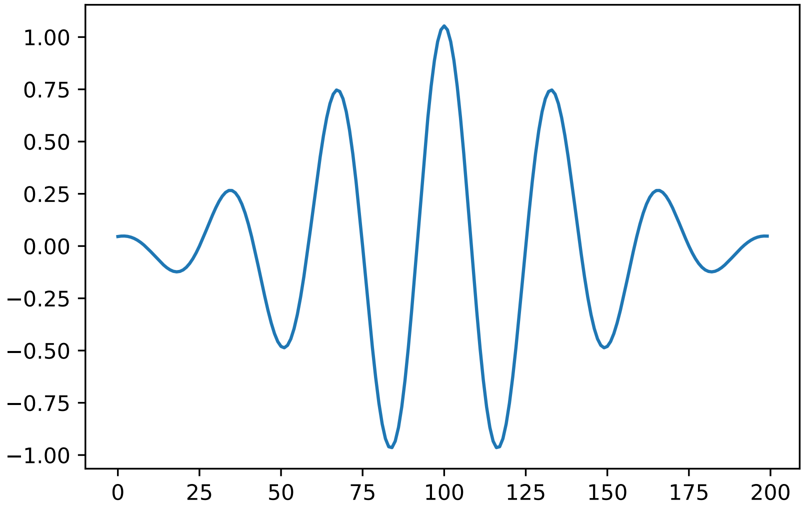


Figure 3.13: One example of real gabor filter. Frequency: 3.0; bandwidth: 0.9492

thermal comfort; this is consistent with the notion that thermal comfort that is "too high" may cause people to become sleepy.

Gabor filtered time series

A 1-D (temporal) Gabor filter is a complex-valued band-pass filter, with a specifiable center frequency and bandwidth, whose impulse response is local in both time and frequency; an example of the real component of one filter is shown in Figure 3.13. Gabor filters have been applied to various facial expression recognition tasks [113] and can capture certain patterns of a raw time series. For instance, they can capture wave-like patterns such as repeated blinking or eye closure. Here, we explored whether they could be helpful for predicting thermal comfort.

Recurrent neural networks

Recurrent neural networks such as LSTM and GRU, are powerful models for dealing with time series. We explored whether a GRU (Gated Recurrent Unit) network can analyze the facial expression series to estimate thermal comfort. We trained a GRU model from the feaures extracted using OpenFace described above using leave-one-person-out cross-validation to measure accuracy of the approach. Hyper-parameters were selected from the sets {learning rate: {0.0001, 0.0005, 0.001}, hidden units: {8, 16, 32}, epoch: 50, optimizer: {Adam, SGD}. For each fold, we randomly selected 5 participants as the validation set (for hyperparameter validation), and the remaining 18 participants as the training set. Training every 5 epochs, the model would be applied to validation set and test set.

After tuning the hyper-parameters on the validation set, the best combination was {learning rate: 0.0005, hidden units: 32, epoch: 15, optimizer: Adam}. The average (over all 24 folds) correlation between predicted and actual thermal comfort scores was 0.248; the result was statistically significant ($p = 0.0425$, Wilcoxon signed-rank test). We note, however, that this result is no larger than the magnitude of the correlation between the eye-lid distance and thermal comfort reported above.

3.6 Discussion and Conclusion

We conducted an experiment in to investigate the relationship between thermal comfort and students' performance in a computer-based learning task in the classroom. We also explored different sensors and predictive models to measure thermal comfort automatically.

Key results: 1) Changing the room temperature by a few degrees Celsius could stat. sig. impact students' self-reported TC; (2) Our experimental data provide evidence that learning is optimal when thermal comfort is neither too high nor too low (inverted U relationship), corroborating prior work. However, we also found a more nuanced relationship than had been identified in prior literature: the impact of thermal comfort on learning was stronger during the third tutorial session (later in time) compared to the first two sessions. (3) Engagement, as labeled by an external observer, was correlated with learning. (4) Thermal comfort can be predicted from the face temperature using an IR camera. (5) Facial expression, at least in the ways we analyzed it, carries only limited information about thermal comfort.

Future work: Given a larger video dataset of face images and associated self-reported thermal comfort scores, we could explore more powerful prediction models that directly predict thermal comfort from the face pixels. This might offer more powerful information than the facial expression estimates from OpenFace.

Acknowledgements: This material is based upon work supported by the National Science Foundation under Grant Nos. #1551594 and #1822768.

Chapter 4

Measuring the effect of ITS feedback messages on students' emotions

Some students don't like a lot of attentions from the teacher or the ITS. Even for the students who are not averse to attention, too much attention may also let the students feel nervous or embarrassed. Feedback messages or hints is one kind of attention. Providing the effective ones to the right students is also an important step for achieve personalization in ITS. In this project, we investigated the effect of the feedback messages on students' emotions in an educational dataset.

4.1 Introduction

One of the main goals of contemporary research in intelligent tutoring systems (ITS) is to promote student learning by both *sensing* the student's emotions and *responding* with affect-sensitive feedback that is appropriate to the student's cognitive and affective state. For sensing students' emotions, a variety of methods are now available, including physiological measurements [83], facial expression analysis [97], and "sensor-free" approaches [66] based on analyzing the ITS logs. Given an estimate of what the student knows and how they feel, the tutor must then decide how to *respond*. Based on the intuition that good human tutors are often empathetic and supportive, many ITS today provide real-time "empathic feedback" to learners that tries to encourage and motivate them to keep learning. This feedback can range in complexity from short utterances [5, 38, 76, 33] to longer prompts [8, 38, 77, 63] such as growth-mindset [24] messages.

Empathic feedback messages could make learners' interactions with ITS more natural and effective, but they also increase the complexity of designing the ITS and its control policy, i.e., how it acts at each moment. Moreover, if feedback is given injudiciously, it could become distracting and suppress learning [33]. While affect-aware ITS with empathic feedback have demonstrated some notable success [44, 97, 8], the sum of evidence of their benefit is unclear. Empathic feedback has often been evaluated as part of a treatment condition in which the feedback was not the only variable being manipulated [76, 8]. Moreover, optimistic hypothesis testing that did not account for multiple hypotheses was often used.

In this paper we investigate the instantaneous impact of ITS feedback on each student's emotional state. The context of our study is an iPad-based system for cognitive skills training [50], specifically a task called "Set" (similar to the classic card game) in which the participants must reason about different dimensions (size, color, shape) of the shapes shown on the cards in order to score a point. The participants are African-American undergraduate students at a Historically Black College/University (HBCU). As measures of emotion, we consider facial expression, heart rate, and heart rate variability, all of which can be estimated automatically, in real time, and with a high temporal resolution.

We examine the following **research questions**: Is there an instantaneous change in facial expression and/or heart rate after each ITS feedback message that is consistent across the participants? Does the evidence for such a change persist even after taking possible confounds into account? Is there evidence that at least *some* participants may exhibit a relationship between the sensor readings and the prompts, even if not all of them do? Finally, is there evidence of any non-emotional change in students' behavior as a result of the feedback messages?

4.2 Related Work

Empathic Virtual Agents: [77] compared an "empathetic" avatar to a "non-empathetic" one. At the start of the experiment, the empathetic avatar would ask the user, "Hopefully, you will get more comfortable as we go along. Before we start, could I please have some of your information?" with the goal of building trust and comforting the participant. In contrast, the non-empathetic one would simply ask, "Have you participated in similar tests before?" They found that the empathetic agent performed no better, in terms of changing students' self-reported mood after the intervention, than the non-empathetic agent. However, they did find in the questionnaire results that participants found the empathetic avatar to be more "enjoyable, caring, trustworthy, and likeable". In another study on virtual agents [85], the researchers compared an "empathic" virtual therapist with a "neutral" one. The empathic therapist was designed to respond to the participant "in a caring manner". For instance, at the start of the session, it would say, "I'm very happy to meet you and hope you'll find our session together worthwhile. Please make yourself comfortable," whereas the neutral therapist would say simply, "Hello, I am Effie a virtual human." The study found that the empathic therapist was beneficial, relative to the neutral therapist, only for a subset of participants; this is reminiscent of the study by [33] who found that the emotionally-adaptive ITS only helped students with less prior knowledge. Moreover, the benefit of the empathic therapist did not persist after the first meeting between the participant and the agent.

Empathic ITS: In [92], the researchers assessed the impact of ITS empathic feedback on students' emotions by manually coding students' facial expressions (frustration, confusion, flow, etc.). They found that there was a difference, in terms of the transition dynamics of students' affective states (e.g., flow to boredom), between the feedback messages that were rated as "high-quality" versus "low-quality" by the students. [38] compared different types of ITS feedback – epistemic, neutral, and emotional – in terms of their impact on facial emotions. The epistemic feedback was more impactful than the emotional feedback in their study. However, their study did not compare to giving no feedback at all. In [63], feedback of different types – growth mindset, empathy, and success/failure – were compared in terms of students' subsequent self-reported emotions. Their results suggest that the different feedback conditions were associated with different emotions (interest, excitement, frustration, etc.). Widmer [120] employed a Wizard-of-Oz experimental design similar to ours to assess the benefit of prompts in ITS; they measured the impact on learning but not on students' emotions.

Multiple Hypotheses: Most prior studies on ITS feedback messages tested many hypotheses but did not statistically correct for this. It is thus possible that they were overly optimistic when identifying possible impacts.

4.3 Sensors of Emotion and Stress

In our work we investigate the impact of ITS feedback on emotion as it is expressed by facial muscle movements and changes in heart rate.

Heart Rate: Heart rate (HR) and heart rate variability (HRV) are well known and widely used as a biomarker of stress [106, 26, 83, 37]. To measure HR and HRV, we use a Polar heart monitor chest belt that is connected wirelessly to a laptop to record the inter-beat-interval (IBI) of heartbeats. We measure HR as the inverse of the IBI, and the HRV as the standard deviation of the IBI.

Facial Expression: Behavioral and medical science researchers have used facial expression as a way of assessing various mental states such as engagement [119], driver drowsiness [31], thermal comfort [55], and students' emotional states in ITS [95]. Facial expression sensor toolkits are now also used in several prominent intelligent tutoring systems

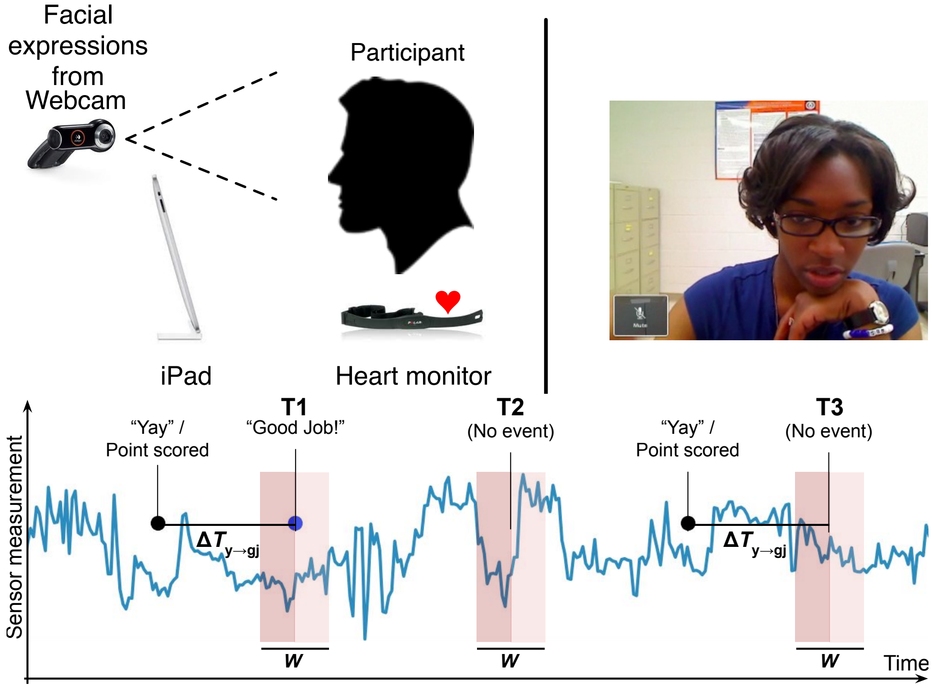


Figure 4.1: Top Left: Experimental setup. **Top Right**: View of the student from the camera. **Bottom**: Methodology: For all message types and sensors (heart rate, facial expressions), we compute the difference in average sensor value $W/2$ sec before vs. after an event (T1), and compare it to the corresponding difference at a random timepoint not near a message (T2). For "Good Job" and "Great Job" messages, to remove a confound due to the "Yay", we compare the difference in average sensor value at T1 to the corresponding difference at another time (T3) that is also $\Delta T_{\mathrm{y}\to\mathrm{gj}}$ sec after "Yay" but not near a feedback message.

[59, 97, 44]. In particular, we use the Emotient SDK from iMotions, which can recognize 20 Facial Action Units (1, 2, 4, 5, 6, 7, 9, 10, 12, 14, 15, 17, 18, 20, 23, 24, 25, 26, 28, 43) [34] and 12 emotions (anger, joy, sadness, neutral, contempt, surprise, fear, disgust, confusion, frustration, positive sentiment, negative sentiment). In each frame, the Emotient SDK could provide a numeric value for each facial expression if there is a face detected.

4.4 Dataset

In our analysis, we examined the HBCU2012 dataset [96] which is an extension of the HBCU dataset from [119]. In HBCU2012, $n = 36$ African-American undergraduate students interacted with iPad-based cognitive skills training software that is designed to strengthen basic cognitive processes such as working memory and logical reasoning. While interacting with the software, their facial expressions and heart rate is being recorded (see Figure 4.1). Each participant interacted with the ITS for 3-4 periods each, resulting in a total of 108 videos.

Procedure: Each student participated for 3-4 sessions, and each daily session lasted about 40 minutes. Although the system contains several tasks, the main task is called Set, which is similar to the classic card game. In this task, the player scores a point if they correctly group 3 cards together that have a correct configuration of size, shape, and color. When the student scores a point, the software automatically issues a "Yay!" sound. The Set task is highly demanding, particularly at the advanced difficulty levels and given the time pressure. At the start of each daily session, the participant takes a 3min pretest. Then, they undergo 30min of cognitive skills training that is facilitated by the system. In particular, the tutor decides the difficulty level at which the student practices, when to switch tasks to take a break, etc. The tutor also issues hints and prompts of different types (described below). During this practice section (but not during the tests), the student receives various feedback messages (see below). After the practice session, the participant takes a posttest.

Types of feedback: The tutor can issue feedback messages of various types (see Table 4.1). Some of them are empathetic, some are motivational, and some are goal-oriented. Note that each message type may be expressed with slightly different phrasing, e.g., "Good Job" might be spoken by the tutor as "Good Job" or just "Good"; "Try harder" can be expressed as either "It seems like you are not trying. Please try your hardest." or "Try harder."

Human-assisted ITS: While in many aspects the cognitive skills training software used to collect the HBCU2012 dataset was automated, the decisions of when to issue feedback messages were made by a human tutor (sometimes called the *trainer* in a cognitive skills training regime) who was either in another room (Wizard-of-Oz style) or in the same room (1-on-1 style) as the participant. For the Wizard-of-Oz setting, the trainer could watch the student's face via a live webcam and also observe the student's practice on a real-time synchronized iPad. Compared with a fully automated ITS, this human-assisted apparatus might actually yield feedback messages that are more appropriately timed and chosen than what an ITS would decide.

Sensor Measurements and Synchronization: Each participant completed the cognitive skills testing and training on an iPad. The inter-beat interval (IBI) of heartbeats was recorded using a Polar heart monitor. Facial expressions were estimated in each frame (30 Hz) of video recorded by a webcam connected to a laptop. The game log was recorded wirelessly from the iPad onto the laptop. Game log, heart rate, and facial expression events were synchronized by finding a common timepoint between the face video and game log.

Prompt	Total Events	Events per Learner: Avg. (s.d.)
Great Job	1950	18.06 (12.72)
Good Job	2935	27.18 (15.89)
Nice Try	621	5.75 (5.46)
Watch Your Time	522	4.83 (2.75)
Keep Going	655	6.06 (4.99)
Faster	1025	9.49 (8.08)
Unique	55	0.51 (1.08)
Different Dims	220	2.04 (2.80)
Brief Directions	88	0.81 (1.09)
Try Harder	45	0.42 (0.98)
Missing	63	0.58 (1.33)
Extra Card	156	1.44 (4.16)
Take Break	33	0.31 (0.57)

Table 4.1: Frequency of the various prompts in our system.

4.5 Methodology

Since all participants received multiple feedback messages, we used a within-subjects design. To assess whether the various messages were associated with any immediate change in students' emotions (see Figure 4.1), we measured the change in the average value of a specific sensor (heart rate, heart rate variability, or one of the 20 AUs + 12 emotions) around the time (T1) when a specific message was issued. Specifically, we computed the average sensor value within a time window of length $W/2$ just after T1 and subtracted the corresponding average sensor value in the time window of length $W/2$ just before T1; this yields Δv. These values, at different times T1, constitute the treatment group of our study. Then, we computed the difference Δv (after-before) at a *random* timepoint (T2) in the participant's time series that was not within 10 seconds of any other prompt. These values, at different times T2, constitute the control group. By comparing Δv due to the treatment vs. the control group, we can estimate the effect of the feedback message on the change in the sensor value. While this is not a truly causal inference approach, our methodology does eliminate the confound that could arise, for example, if the average sensor value tended to increase (or decrease) over time, e.g., due to fatigue.

Repeated Measures Design: Since we have multiple feedback messages and multiple days of participation for each student in our study, we use a repeated-measures design based on a linear mixed-effect model, where the student ID is a random effect. We then assess whether the presence (1) or absence (0) of the feedback message is statistically significantly related to the change Δv in a specific sensor value (facial expression or heart rate value). We repeat this for all message types and sensor values.

Hypothesis Correction: Due to many hypotheses (different messages and sensor measurements) that are largely independent of each other and lack of strong prior belief that a relationship exists between any particular sensor and feedback message, we take a conservative approach and perform Bonferonni correction to the p-values: Instead of the traditional $\alpha = 0.05$ threshold, we require $\alpha = 0.05/m$, where m is the total number of hypotheses.

Effect Size: We quantified the effect size in two ways, both of which are a form of Cohen's d statistic: (1) Global effect size: we divided the fixed-effect model coefficient for the treatment by the standard deviation of the sensor value (e.g., happiness value) over the *entire dataset* (all participants, all days, and all times). (2) Local effect size: we divided the fixed-effect model efficient for the treatment by the standard deviation of all Δv in the union of the treatment and control groups. This expresses whether the change due to the feedback message is large compared to changes that occur in other time windows of length W.

4.6 Analysis

4.6.1 Facial Expression

Analysis Details: We followed the methodology described above, where we picked 20 time points (T2) per each video such that there are no other event 10s before or after them for the control group. For the time window W, we used 5s and 10s. We allowed for the possibility that the participants' reactions to the ITS feedback messages might be slightly delayed; hence, we conducted analyses with a "right-shift" parameter τ of either 0s or 1s. Finally, for the number of hypotheses m by which we corrected the p-value threshold α, we considered that the 12 *emotions* (happy, sad, angry, etc.) can be considered combinations of individual Facial Action Units (AUs) [34] and are thus not independent of the 20 AUs we already measure. Since there are 13 different ITS feedback messages that we consider, we thus let $m = 13 * 20 = 260$ so that our threshold α for statistical significance by Bonferonni correction is $0.05/260$.

Results: Only 2 of the 13 feedback messages showed any stat. sig. impact, after p-value correction, on *any* of the 32 facial expressions for any of the right-shift values (0s, 1s) or window sizes (5s, 10s). The two message types were "Great Job" and "Good Job", and the effects were significant across all combinations of W and τ. Table 4.2 show the facial expression values that have a significant change due to these feedback messages. Note that the effect sizes are generally quite small, especially when assessed at a global level (i.e., relative to the variance of the expression value over the whole dataset). The largest

	Great Job			Good Job		
Facial Evidence	**p-value**	**Global Effect Size**	**Local Effect Size**	**p-value**	**Global Effect Size**	**Local Effect Size**
Fear	2.59e-12	0.045	0.105	4.6e-10	0.077	0.092
Disgust	9.38e-09	-0.044	-0.197	1.33e-13	-0.104	-0.245
Sadness	6.06e-05	-0.023	-0.081	6.06e-05	-0.023	-0.081
Confusion	8.91e-05	-0.030	-0.150	2.48e-06	-0.062	-0.113
Neutral	-	-	-	5.9e-05	-0.067	-0.160
AU1	-	-	-	5.42e-06	0.035	0.074
AU4	6.75e-08	0.018	0.074	2.71e-07	0.032	0.068
AU5	<2e-16	0.08	0.344	<2e-16	0.120	0.335
AU7	7.11e-08	0.022	0.121	9.87e-12	0.045	0.10
AU15	-	-	-	1.30e-04	0.036	0.08
AU18	-	-	-	3.65e-07	-0.060	-0.106
AU20	7.39e-05	0.025	0.106	4.87e-05	0.034	0.041
AU25	-	-	-	1.60e-04	0.047	0.128
AU26	-	-	-	9.33e-05	0.038	0.148
AU43	<2e-16	-0.086	-0.350	<2e-16	-0.156	-0.736

Table 4.2: "Great Job/Good Job": Effects on facial expression values which are stat.sig. for $W = 10\text{s}$, $\tau = 0\text{s}$.

absolute effect size is for AU43 (closing of the eyes) for both "Good Job" and "Great Job", whereby the participants' eyes tend to be more closed before than after the message.

4.6.2 Heart Rate

Analysis Details: We varied W over 5s and 10s, and the trends were the same. For Bonferonni correction, we let $m = 26$ since we considered two different heart measures (HR, HRV) and there were 13 different message types.

Results: None of the prompts showed a stat. sig. impact on HR or HRV.

4.7 Effects on Individual Students

Here we consider the hypothesis that the feedback messages may affect *some* students but not others. In particular, we test, for each combination of participant, feedback message, and sensor measurement, whether there is a statistically significant difference *within each student* in the average sensor value $W/2$ seconds after vs. before the prompt. For each combination of prompt and sensor value, we then calculate the fraction of students for which the difference is statistically significant. Importantly, this analysis allows for a different effect – some positive, some negative – on each student.

Facial Expression: We perform the analysis for $W = 10\text{s}$. If, for each student, *any* of the 32 facial expression values were significantly changed due to a feedback message, then we increment our count for that message type. We let m (number of hypotheses) be 20 (the number of unique Facial Action Units we measure) and hence $\alpha = 0.05/m$ =2.5e-03. The results shows that for most messages, less than one quarter of the students showed any effect; only the "Great Job"(18/36) and "Good Job"(19/36) affected at least half of the students

Heart Rate: We varied W over 5s and 10s, and the trends were similar. For Bonferonni correction for each participant, we let $m = 2$ since we considered two different heart measures (HR, and HRV). The trend is similar as for the facial expression measures ("Great Job": 16/36; "Good Job": 19/36).

4.8 Impact of "Great Job" and "Good Job" Messages

Our analyses have found robust (over multiple sensor measurements, right-shifts, and window sizes) evidence of a relationship between the "Great Job" and "Good Job" messages and facial expression (but not heart rate), despite the conservative Bonferonni correction. However, there was little evidence in support of any other feedback message. Given that these two message types almost always occur shortly after the student has scored a point, we explored whether the change due to the feedback itself or simply because the point scored a point. To examine this, we modify the methodology from Section 4.5 so that the control group for these messages is taken at times T3 that are $\Delta_{\mathrm{y}\to\mathrm{gj}}$ after a "Yay"/point scored timepoint but where no such feedback occurs (see Figure 4.1). Importantly, the decision of whether or not "Good Job"/"Great Job" was given was at the discretion of the human trainer and was essentially random (i.e., quasi-experimental analysis). This allows us to isolate the effect of the feedback itself, rather than of the preceding "Yay" sound. We estimated the value $\Delta_{\mathrm{y}\to\mathrm{gj}}$ over all the "Great Job" and "Good Job" messages in our dataset (around 1.091s).

Analysis Details: We selected "Great Job" and "Good Job" timepoints T1 such that there is no other message before and after 5 seconds except a "Yay". We also randomly selected a similar number of time points for T3. We varied W as 5s or 10s, and we let τ be 0s or 1s. Since there are now just 2 feedback messages and 20 AUs, we let $m = 40$.

Results: After accounting for the preceding "Yay"/point-scored as described above, we find *no* statistically significant change of any facial expression before vs. after the "Great Job" or "GoodJob", for any W or τ. This indicates that the change in facial expression around these messages is likely due to having scored a point, not the feedback itself.

4.9 Conclusions

Our analyses of facial expression and heart rate data from 36 African-American students interacting with iPad-based cognitive skills training software suggest that (1) the impact of the short empathic feedback messages on students' emotions was very small. (2) Several of the correlations (for "Good Job" and "Great Job") disappeared after we accounted for the confound that the student's own achievement at having scored a point could explain the impact. (3) When examining the emotional impact on *individual* students, we found that, except for "Great Job" and "Good Job", only a modest fraction of students showed any stat. sig. correlation. Therefore, before trying to optimize an empathic ITS' control policy, it may be worth verifying that the feedback messages have any impact at all. On the other hand, and more optimistically, contemporary emotional recognition systems also offer a pathway forward to measure the impact of the ITS' actions more precisely. Finally, we note that there could be non-emotional effects of the ITS prompts on students' behaviors. For instance, when watching some videos, we noticed that a few participants shifted their eye gaze in response to the "Watch your time" prompt. Future work can explore this issue.

Acknowledgments: This research was supported by the NSF National AI Institute for Student-AI Teaming (iSAT) under grant DRL 2019805, and also by an NSF Cyberlearning grant 1822768. The opinions expressed are those of the authors and do not represent views of the NSF. a

Chapter 5

Can the Mathematical Correctness of Object Configurations Affect the Accuracy of Their Perception?

Object detector can be very useful in educational content searching, captioning and indexing. Improving the accuracy of the detector can provide a better summarization to the video by the detected objects. In this project, we proposed a new kind of dataset bias and investigated that could this bias influence the accuracy of detection results on the images with math expressions.

5.1 Introduction

Visual context affects object perception. Extensive research in psychology and neuroscience on human perception, as well as computer vision and machine learning research on artificially trained models, has demonstrated how the context can impact perception in both detection and recognition tasks (e.g., [79, 14, 86]). In human perceivers, the mechanisms of how surrounding objects can affect object perception include modulated visual attention as well as changes to how low-level features are integrated when forming high-level judgments about object categories [79]. Within the computer vision community, this line of research has partly motivated the collection of new datasets (e.g., CLEVR) so as to reduce biases in their ground-truth labels that could otherwise be exploited by trained models to obtain a deceptively high accuracy [58].

To date, work within machine learning and computer vision on dataset bias has focused mostly on statistical *correlations* between an object and its context that can be learned during training, e.g., if a training dataset contains boxes that are mostly red, then that statistical dependency can affect perception at test time as well. However, a related and arguably deeper question is whether a neural network's accuracy could be influenced by its understanding, or lack thereof, of *semantic* relationships between objects and their attributes that *generalize* beyond mere co-occurrence.

As a specific motivating application that we recently encountered, suppose one wishes to train an object detector to find all the math content (expressions and equations) within each frame of a collection of math tutorial videos, so that the math content in the videos can be more easily searched. Might a CNN trained on such videos learn a bias whereby the *correct* content (e.g., "$5 - 2 = 3$") is more likely to be detected as a visual object than incorrect content (e.g., "$5 - 2 = 4$")? Could such a bias be learned by generalizing the rules of arithmetic beyond the finite set of examples that were provided during training (in other words, can the machine implicitly learn that $5-2=3$ even if this specific combination of operators and operands was not part of the training set)? For another example, suppose a computer vision-based automatic homework grading system (e.g., as developed by the company GradeScope) was trained to evaluate whether each student solved a set of algebra problems correctly; would the system suffer in accuracy of detecting *individual symbols* if it was trained only on examples of *correct* solutions, in which the configurations of symbols followed the rules of algebra?

On the surface, it may seem obvious that a network trained on a dataset with some property $P=1$ (e.g., whether all the equations rendered in the images are mathematically correct) should do better when tested on a dataset for which $P = 1$ compared to when $P=0$ (e.g., the equations rendered in the images are often incorrect). This would be especially so if P could be learned by the model through memorization of specific objects from training images, or if P were based on simple correlations such as "if object X appears, then object Y usually also appears". However, we argue that the answer to the question is not obvious when the bias is based on non-trivial semantic (rather than just correlational) relationships between objects (e.g., subtraction of two-digit numbers requiring "borrowing" from the 10's to the 1's place) that might be difficult for a neural network to learn even with explicit training, and when the machine must generalize to novel combinations of objects never seen during training.

In this paper, we describe a sequence of experiments to explore the influence of the mathematical correctness of object configurations in a visual scene on the accuracy of their perception by simple CNN architectures.We investigate the effect in both object recognition and object detection tasks, and in two different settings: mathematical expressions and equations rendered as images, and physical simulations of moving particles. At a high level, our results indicate that (a) neural networks are capable of learning simple mathematical relationships implicitly from how the objects appear together in images, without explicit supervision of what the objects mean or what the relationships are; and (b) the dataset bias, in terms of the mathematical correctness in the configurations of objects, can affect the network's perception accuracy at test time – even on specific configurations of objects never seen during training. Our paper contributes to the growing interest in dataset bias, as well as on causal models [98] that are valid beyond the standard "in-distribution generalization" paradigm [3].

5.2 Related Work

Bias in Neural Network Training: One common weak point of neural networks is that they easily overfit to biases in the training data. [2] points out that in the Visual Question Answering (VQA) dataset [6], just because a model can correctly answer some image-question pairs does not necessarily mean the model is trained well, due to the possibility of label bias in the training dataset. For example, a model might be trained to answer the question, "What covers the ground?" in a dataset in which snows always covers the ground. For the goal of helping trained models to generalize better, [58] created a new dataset (CLEVR) that minimizes the kinds of questions that do not require actual visual reasoning, thereby reducing the bias caused by the co-occurrence of two objects in the image.

In the domain of object recognition, [41] showed that CNNs trained on ImageNet [27, 101] often use textural information more than shape information. In their example, a cat with Indian elephant texture was recognized as an "Indian elephant" rather than a "cat". This kind of bias might be caused by the uniqueness of the texture of that class. In each image, the Indian elephant can have multiple shapes and poses, but almost all the textures are the same, and the texture is often easier for the network to harness for the recognition task. In [3], the authors described how non-semantic features such as color can influence the network's output. They proposed four different kinds of training regimes: in-distribution generalization, generalization under non-systematic-shift, generalization under systematic-shift and semantic anomaly detection.

Learning Mathematical Relationships: A number of works [22, 78, 116] have investigated the extent to which neural networks can be trained to solve mathematical problems. However, relatively little prior literature has explored whether neural networks can learn mathematical logic from images directly, rather than via explicit supervision. For example, [51] used two images that contained numbers as the input to a feed-forward neural networks and an image that contained the results of the two input numbers as the output. The operations could be addition, subtraction or multiplication. There was no extra information about what the characters (numbers) mean to the model. Their results showed that some mathematical concepts (addition and subtraction) could be purely learned by visual information. [71] presented a CNN based model that could learn to perform addition using input images that contained a mathematical expression, e.g., "$6 + 9$", without knowing what the characters "6" and "9" mean in advance. In [47], the authors defined a mapping from the Fashion MNIST to "0" to "9", and used this mapping to generate a new math dataset which used the Fashion MNIST examples as the numbers. The input to the model (RNNs or CNNs) were two images, and the output was also an image that contained the result of the input numbers. Their results showed that bitwise-and and bitwise-or were easier to learn than addition and subtraction. Finally, [126] found that CNN-based models also have the ability to learn some cognitive reasoning tasks such as symmetry, counting, etc. They found that, while humans can learn the tasks from just a few examples and achieve 100% accuracy after humans mastered this task, the neural networks require a large number of training examples and and cannot "master" the tasks like humans can.

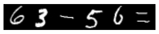
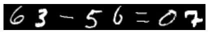
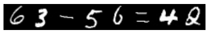
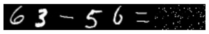
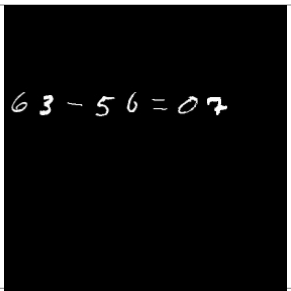
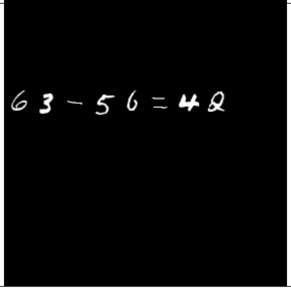
5.3 Experiment I: Learning to Perceive Subtraction Problems

In our first experiment, we assessed whether the mathematical correctness of object configurations that are rendered as images affects the accuracy in recognizing or detecting the equations' individual objects. Here, the "objects" are symbols (0-9, -, =) that describe a mathematical equation, and the mathematical relationship between the symbols is the subtraction operation (which requires "borrowing" from the 10's to the 1's place). Because we were uncertain at the onset as to whether a CNN could implicitly learn the mathematical relationships implicitly (i.e., without supervision of the correctness) and directly from pixels, we conducted the experiment in a sequence of stages of increasing difficulty.

Dataset: We considered math problems of the form $a - b = c$ and generated the set of all $n = 99 * (99 + 1)/2 = 4950$ unique tuples $\mathcal{T} = \{(a, b, c) \in \{0, 1, \ldots, 99\}^3 \mid (a > b) \wedge (c = a - b)\}$. All tuples in dataset $\mathcal{T}$ are *mathematically correct*. For instance, $\mathcal{T}$ contains the tuple $(55, 23, 32)$ since $55 - 23 = 32$. We then partitioned $\mathcal{T}$ into training, validation, and testing subsets ($\mathcal{T}^{\text{tr}}, \mathcal{T}^{\text{va}}, \mathcal{T}^{\text{te}}$, have 2476, 1237, and 1237 examples, respectively) such that, if (a, b, c) occurs in one subset s, then (a, c, b) must also occur in the same subset. The purpose of the latter condition was to ensure that, in order to achieve high accuracy, the network must learn the full semantics of the mathematical operation of subtraction, and not perform well just by harnessing the (relatively) simple rule that $a-b = c \implies a-c = b$. Our goal in designing this dataset and its subsets was to ensure that the network has to learn to generalize to new math problems entirely, not just novel images of previously seen same math problems.

Since we were interested in how the dataset bias of mathematical correctness can af-

Table 5.1: The various datasets and tasks we used to train the networks in Experiment I on learning to perceive mathematical equations.

	Network Input		**Target**	
#	Description	Example ($63 - 56 = 07$)	Description	Example
1	Image of $a_1a_2 - b_1b_2 =$		One-hot codes of c_1, c_2	$[1, 0, 0, 0, 0, 0, 0, 0, 0, 0]$, $[0, 0, 0, 0, 0, 0, 0, 1, 0, 0]$
2	Image of $a_1a_2 - b_1b_2 = c_1c_2$		One-hot codes of a_1, a_2, $-, b_1, b_2, =, c_1, c_2$	$[0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0]$, $[0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0]$, $\dots$ $[0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0]$
3	Image of $a_1a_2 - b_1b_2 = c'_1c'_2$		One-hot codes of a_1, a_2, $-, b_1, b_2, =, c'_1, c'_2$	$[0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0]$, $[0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0]$, $\dots$ $[0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0]$
4	Image of $a_1a_2 - b_1b_2 =$(noise)		One-hot codes of a_1, a_2, $-, b_1, b_2, =, c_1, c_2$	Same as #2
5	Image with $a_1a_2 - b_1b_2 = c_1c_2$ as subimage		Bounding boxes of a_1, a_2, $-, b_1, b_2, =, c_1, c_2$	(0,28,28,28), (28,28,28,28), $\dots$
6	Image with $a_1a_2 - b_1b_2 = c'_1c'_2$ as subimage		Bounding boxes of a_1, a_2, $-, b_1, b_2, =, c'_1, c'_2$	(28,140,28,28), (56,140,28,28), $\dots$

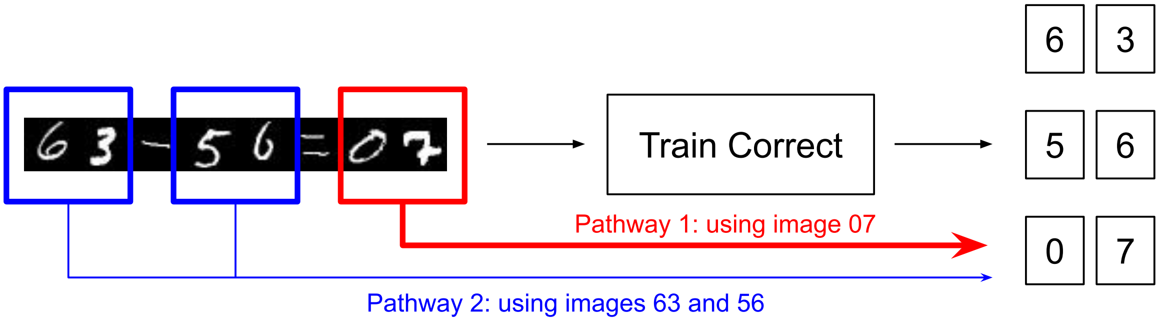


Figure 5.1: Two alternative pathways for how the "Train Correct" network can classify the symbols in mathematically correct expressions.

fect the perception accuracy, we thus also generated a dataset of *random* tuples $\widetilde{\mathcal{T}}^s = \{(a^{(i)}, b^{(i)}, c^{(\sigma(i))}) \mid (a^{(i)}, b^{(i)}, c^{(i)}) \in \mathcal{T}^s\}_{i=1}^n$, where $s \in \{\text{tr}, \text{va}, \text{te}\}$, and σ is a permutation of indices $1, \ldots, n$. Naturally, the vast majority of these will be mathematically incorrect (specifically, only 1.45%, 1.30% and 1.78% of the tuples were correct in the training, validation and testing subsets, respectively). Since the marginal probability distributions (within each of the training, validation, and testing subsets) $P(c)$ are the same in both $\mathcal{T}^s$ and $\widetilde{\mathcal{T}}^s$, the baseline accuracy of just guessing the majority class for c in the test set is also the same.

Hypotheses: A computer vision model trained on mathematically correct data can recognize the digits of c using two alternative pathways (see Figure 5.1): (1) perceive c from its pixels; or (2) predict c by perceiving a and b and then subtracting them. A network trained on random data, on the other hand, can only use pathway (1). Hence, we hypothesize that the following relationships about the symbol recognition accuracy (averaged over all 8 symbols in each equation) will hold:

1. *Train Random, Test Random* $=$ *Train Random, Test Correct*. If the network is trained on $\widetilde{\mathcal{T}}^{\text{tr}}$, then the symbol recognition accuracy is independent of the mathematical correctness of the relationship between (a, b) and c.

2. *Train Random, Test Random* $>$ *Train Correct, Test Random*. The network trained only on correct tuples will suffer when it is tested on random tuples, since pathway (2) above will usually be misleading.

3. *Train Correct, Test Correct* $>$ *Train Correct, Test Random*. Same reason as hypothesis 2.

4. *Train Correct, Test Correct* $>$ *Train Random, Test Correct*. The network trained only on correct tuples will benefit from being able to rely on two alternative pathways (instead of just one), especially when the images of the digits are unclear or are noisy.

5.3.1 Stage 1: Learning to Subtract Numbers

We first wanted to verify that a CNN that receives an image of a novel (i.e., not seen during training) two-digit subtraction problem $a - b$ can correctly compute the answer c with high accuracy. We represented each number (a or b) using two digits (e.g., a is rendered as a_1a_2) by including a leading 0 if necessary. For each digit, we randomly sampled an MNIST image of the appropriate class and concatenated them (along with an = symbol) to produce an image of the form $a_1a_2 - b_1b_2 =$. (See line #1 in Table 5.1.)

Methods: We trained a simple CNN on $\mathcal{T}^{\text{tr}}$ (and $\mathcal{T}^{\text{va}}$ for early stopping) with 4 convolutional layers followed by 2 dense layers (50 neurons each) with batch normalization and dropout using SGD (lr=5e-3). Accuracy was measured as the fraction of examples in which *both* digits of $c = c_1c_2$ were correctly predicted by the network.

Results: When tested on $\mathcal{T}^{\text{te}}$, the network achieved 90.90% accuracy. For comparison, the baseline accuracy for just guessing the majority class in the test set was 2.9%. (Note that the distribution $P(c)$ in $\mathcal{T}$ is not uniform, since there are more tuples (a, b) that yield small c than those that yield large c.) While not perfect, this network provides a proof-of-concept that a network can learn the subtraction operation with high accuracy on subtraction problems $a - b$ not seen during training.

5.3.2 Stage 2: Recognizing Digits in Equation Images

Next we investigated whether the mathematical correctness of the tuples (a, b, c) used to train and/or test the neural network affects the accuracy of the CNN in recognizing all the individual symbols (0-9, -, =) in images of the form $a_1a_2 - b_1b_2 = c_1c_2$. The networks we train have the same architecture as in Stage 1, except that the network takes an input image of size 28×224 (since there are 8 input symbols in total) and produces 8 different one-hot vectors (with 12 elements each for 0-9, -, and =) as output. We conduct a 2x2 experimental design: we train the network on either mathematically correct equations or on random data; and then we test the network on either mathematically correct or on random data.

Methods: We trained the network (SGD with lr=3e-4) on *either* mathematically correct examples $\mathcal{T}^{\text{tr}}$ *or* random examples $\widetilde{\mathcal{T}}^{\text{tr}}$. (See lines #2 and #3 in Table 5.1; note that c'_1 and c'_2 refer to the two digits of a c from a *random* example in $\mathcal{T}$, as described in the beginning of Section 5.3.) We then tested each of the two networks on either $\mathcal{T}^{\text{te}}$ or $\widetilde{\mathcal{T}}^{\text{te}}$ and compared accuracy within the 2x2 experimental design matrix. We trained the networks using 5 random seeds and averaged the accuracy results to reduce variance.

Results: Accuracy numbers are shown in Table 5.2. We use t-tests (paired or unpaired, depending on the comparison) to check for statistical significance. We evaluate each of our four hypotheses below:

Table 5.2: Mean digit recognition accuracy (std.dev.) in subtraction problem images ($a - b = c$)

	Test Correct	Test Random
Train Correct	96.65% (0.185%)	93.44% (0.177%)
Train Random	95.46% (0.192%)	95.46% (0.208%)

1. *Train Random*, *Test Random* is not stat. sig. different ($p = 0.9062$) from *Train Random*, *Test Correct*; this supports hypothesis (1).

2. *Train Random*, *Test Random* is stat. sig. higher ($p = 5.467 \times 10^{-7}$) than *Train Correct*, *Test Random*; this supports hypothesis (2).

3. *Train Correct*, *Test Correct* is stat. sig. higher ($p = 8.017 \times 10^{-8}$) than *Train Correct*, *Test Random*; this supports hypothesis (3).

4. *Train Correct*, *Test Correct* is stat. sig. higher ($p = 1.954 \times 10^{-5}$) than *Train Random*, *Test Correct*; this supports hypothesis (4).

In sum, these results indicate that, despite imperfect learning of the subtraction operation (90.90% accuracy from Stage 1), the dataset bias of the training set in terms of the mathematical correctness of the object configurations can still impact testing accuracy of individual symbol recognition.

5.3.3 Stage 3: Noisy c

In a follow-up experiment to understand better the results in Stage 2, we investigated what happens to the networks' predictions when the image of c is replaced entirely by noise. In particular, we conducted an experiment using the images rendered from $\mathcal{T}$ only (i.e., mathematically correct expressions), except that – during *testing* – the subimage corresponding to the two symbols in $c = c_1c_2$ was replaced by pure noise. (See line #4 in Table 5.1.) We compared two models: *Train Correct, Test Noisy* c and *Train Random, Test Noisy* c.

Methods: Same as Stage 2, except that we computed test accuracy on just the two symbols in $c = c_1c_2$.

Results: For *Train Correct*, the mean accuracy (std. dev.) was 3.06% (0.265%), whereas for *Train Random*, the accuracy was 1.85% (0.563%); the difference is stat. sig. ($p = 2.04 \times 10^{-4}$). These results further support the hypothesis that the model *Train Correct* can harness two prediction pathways. It also underlines how the dataset bias can be beneficial: if it is known a priori that the images at test time will always be mathematically correct, then the network can be made more robust (in terms of individual symbol recognition accuracy) to noise if it is trained on only correct data.

Table 5.3: Digit Detection Accuracy (mAP) in subtraction problem images ($a - b = c$)

	Test Correct	Test Random
Train Correct	98.94% (0%)	98.30% (0.120%)
Train Random	98.26% (0.05%)	98.26% (0.05%)

5.3.4 Stage 4: Detecting Digits in Larger Images

In our last experiment on perception of images of two-digit subtraction problems, we extended the task to object *detection*: Specifically, we train neural networks both to locate and to classify every symbol (0-9, - =) in the rendered image. (See lines #5 and #6 in Table 5.1.)

Methods: We used YOLO (v1) [88] as the object detection architecture. In contrast to the original network design, the YOLO in our experiments predicted exactly 1 symbol per grid cell. We generated images containing a random equation from $\mathcal{T}$ or $\widetilde{\mathcal{T}}$ placed onto a random location in a black 280×280-pixel background. When generating the images, each symbol was always placed in the middle of a YOLO grid cell.

We trained the networks using SGD (lr=1e-4), batch size of 100, for a maximum of 5 epochs using early stopping. Mean average precision (mAP) was used as the accuracy metric. We trained 2 instances of each network (*Train Correct*, and *Train Random*) to enable statistical significance testing.

Results: Mean average precision values are shown in Table 5.3. Similar to Stage 2, we find support for hypotheses (1), (3), and (4) because the differences between the corresponding pairs of cells in the table were statistically significant. However, in contrast to Stage 2, there was no statistically significant difference between *Train Random*, *Test Random* and *Train Correct*, *Test Random*.

5.4 Experiment II: Learning to Perceive Algebra Problems

In the next experiment we went beyond simple subtraction and explored whether mathematical correctness bias could affect individual symbol recognition accuracy in algebra problems of one variable.

Dataset: The algebra problems were all of the form $pa + q = r$, where a is the variable to be solved, each constant $p, q,$ is an integer between -9 and $+9$ (inclusive), with the additional constraint that the solution $a = (r - q)/p$ was required to be an integer between -5 and +5 (inclusive). From each tuple (p, q, r) representing an algebra problem, we generated images containing two lines of content: The first line represented the equation, whereby the symbols in the rendered equation were randomly commuted according to standard rules of algebra (e.g., the problem $pa + q = r$ was sometimes rendered as $q + pa = r$,

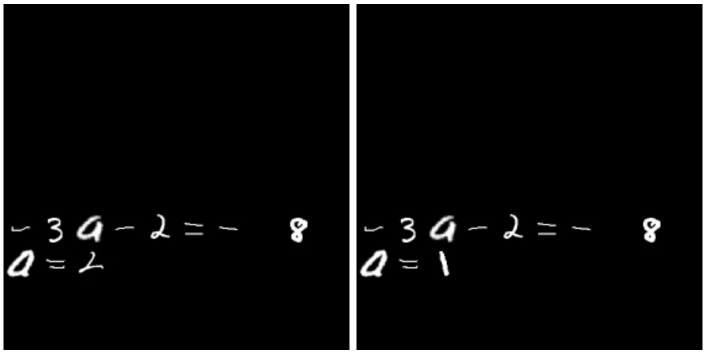


Figure 5.2: Examples of mathematically correct (left) and incorrect (right) algebra problems (Experiment II), where "correctness" is defined in terms of consistency between the putative solution in the second line to the problem statement in the first line.

Table 5.4: Detection accuracy (mAP) in the algebra problem images.

	Test Correct	Test Random
Train Correct	$98.56\%(0.007\%)$	$98.56\%(0.014\%)$
Train Random	$98.58\%(0.113\%)$	$98.58\%(0.113\%)$

$r = pa + q$, or $r = q + pa$); all re-orderings of the same equation were attributed to the same algebra problem (p, q, r) and were always placed into the same data fold (train, validation, test) to avoid data leakage. The second line represented a putative solution to the algebra problem in the form $a = c$. In mathematically correct algebra problems, the value of c equals the true answer $(r - q)/p$. In random problems, c was picked uniformly at random from -5 to $+5$ (this resulted in 13.03%, 12.06% and 14.06% of the solutions being correct in the training, validation and testing subsets, respectively).

Methods: Analogously to Stage 2 of Experiment I, we trained a YOLOv1 to detect and recognize every digit of algebra problems that were placed onto larger images; see 5.2. We used a 2x2 experimental design, as before: { *Train Correct*, *Train Random* } × { *Test Correct*, *Test Random* }. For each of the 4 conditions, we trained 2 models for statistical significance testing.

Results: Results are shown in Table 5.4. In short, there was virtually no difference between conditions. It seems that the YOLOv1 detector was not able to learn the algebraic relationship between p, q, r and the solution for a to high enough accuracy so as to influence the detection accuracies.

Table 5.5: Particle detection accuracy (mAP) in the Colliding Particles experiment.

	Test Correct	Test Random
Train Correct	99.01%	80.74%
Train Random	98.88%	98.89%

5.5 Experiment III: Learning to Perceive Moving Particles

In our final experiment, we switch to a new task to explore whether the trends we found when perceiving subtraction problems also occur in a different setting: physics simulations of moving and colliding particles.

Dataset: We simulated the positions of two particles (one yellow, one red) at three equally spaced timesteps ($t = 0, 1, 2$sec). In the mathematically correct dataset (which we call $\mathcal{P}$), the particles (radius of 4.5 pixels, with starting position chosen uniformly at random between 4 and 45 pixels along each axis) both initially move at a constant speed (chosen uniformally at random from 4 to 12 pixels/sec) toward each other; if and when they collide within the 2 second interval, their collision conserves both momentum and kinetic energy. Each image in $\mathcal{P}$ is the concatenation of the renderings (each 50×50 pixels with 3 color channels) of the particles (plus some random background noise) at the three timesteps. Figure 5.3 (top) shows an example of an image in $\mathcal{P}$. In contrast, the random dataset $\widetilde{\mathcal{P}}$ contains a mixture of images, half of which are correct (drawn from $\mathcal{P}$) and half of which are incorrect (whereby the coordinates of the balls at the three timesteps are generated randomly).

Methods: We trained a YOLOv1 to detect each the 6 particles in each input image, similar to Section 5.3.4. Since training was slow, we trained just one neural network for each experimental condition.

Results: Table 5.5 shows the mean Average Precision (mAP) of the networks *Train Correct* and *Train Random* evaluated on either *Test Correct* or *Test Random*. Figure 5.4 shows examples of object detections. The results are consistent with all four of our hypotheses from Section 5.3. Figure 5.4 shows examples of the detections for the different conditions.

5.6 Conclusions

We have conducted object recognition and object detection experiments, on images of arithmetic (subtractive) expressions, algebra problems, and colliding particle simulations, to explore whether dataset bias regarding the mathematical correctness/incorrectness of the object configurations can impact the accuracy of the objects' perception. For the subtraction problems and particle simulations, we found that the neural networks were, with

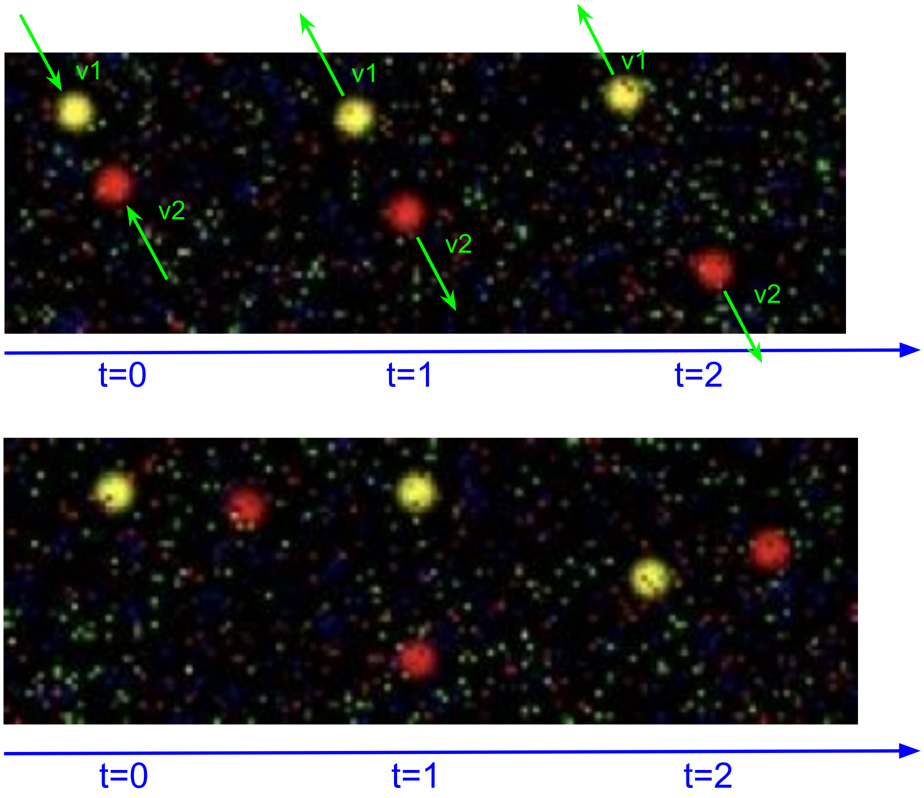


Figure 5.3: Examples images in in the moving particles experiment. Top: mathematically correct example, along with superimposed arrows (for the reader, not rendered in the actual dataset) showing the initial velocities of the particles. Bottom: random (and mathematically incorrect) example.

enough training data, capable of implicitly learning the semantic relationship and generalizing to new scenes containing instances of the relationship that never were seen during training; moreover, the implicitly learned semantic rules yielded small but reliable accuracy differences when tested on a dataset with a different semantic bias. On the algebra problem task, no such effect was observed, possibly because the semantic relationship was too challenging for the network to infer implicitly and directly from pixels; it is however conceivable that more powerful recognition architectures might still be able to learn the relationships.

Importantly, our results go beyond mere correlational label bias and instead address the semantic question of whether high-level relationships can be generalized and influence the network's perceptual accuracy. To our knowledge, these results are the first to demonstrate how mathematical relationships between objects can be learned and influence perception accuracy.

On one hand, our results suggest that, if it is known ahead of time that *all* data at

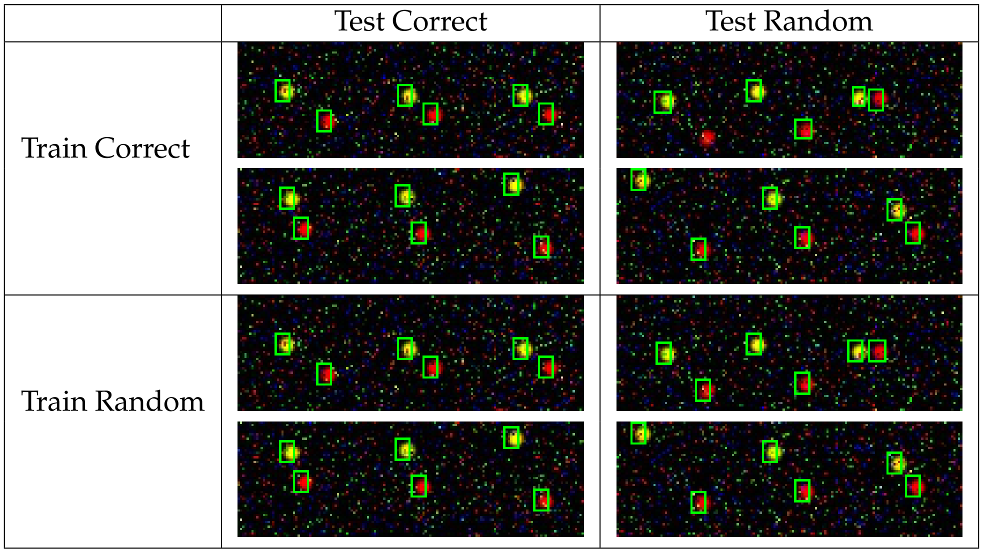


Figure 5.4: Examples of object detections in the moving particles experiment.

test time will adhere to certain mathematical constraints, then it is worth optimizing the network on exactly the same constraints at training time, as this may yield an accuracy advantage. On the other hand, if the mathematical correctness of the objects' configurations can vary from both correct to incorrect, then it may be useful to train the network accordingly. Particular for educational applications on automatic math problem grading, which served as a concrete motivation for this paper, it may be important to assure the students, teachers, and parents, that the recognition accuracy of the *individual symbols* in students' submissions is the same for everyone, regardless of whether their overall math solution was right or wrong.

Acknowledgements: This research was supported by the NSF National AI Institute for Student-AI Teaming (iSAT) under grant DRL 2019805. The opinions expressed are those of the authors and do not represent views of the NSF. This research was also supported by ONR grant N00014-18-1-2768, and by NSF CAREER grant 2046505.

Chapter 6

Representation of Math Videos using Detected Math Information

A good searching and recommendation system for educational videos is important to students when they are learning online. For the long term goal to provide personalized searching and content recommendation, we explored whether computer vision based methods could be used to improve the searching results according to an input math query.

6.1 Introduction

Online learning has steadily gain popularity over past years and become a new normal due to the Covid-19 pandemic [124]. Tutorial videos are ones in which a tutor is explaining something to one or several students. Learning from online tutorial videos is different from in-person tutorial sessions: The students cannot interrupt the tutor and ask questions during the lecture and also cannot ask questions when the lecture ends. Sometimes they need more detailed explanation to understand the learning materials better. Even though numerous tutorial resources can be found online, it is not easy to for students to find the right one. Since anyone can upload material to the internet, the quality of these resources is not always great. The visibility of the video could be poor and the tutors could also provide wrong answers. Students need to spend a lot of time to find the most useful videos. In addition, some videos contain multiple problems. The tutors may not put all the math problems in the title or video description. Using just the video title or description to match the search keywords might miss a lot of relevant content.

Inputting a detailed query, i.e., math expression, to a search engine is similar to asking a personalized question to a tutor in some sense. If the returned videos can explain the question well, the self online learning can be similar to a 1-on-1 in-person tutoring.

Improving the returned video quality can decrease the students' searching time and thus allow them to spend more time on the parts that are not understood.

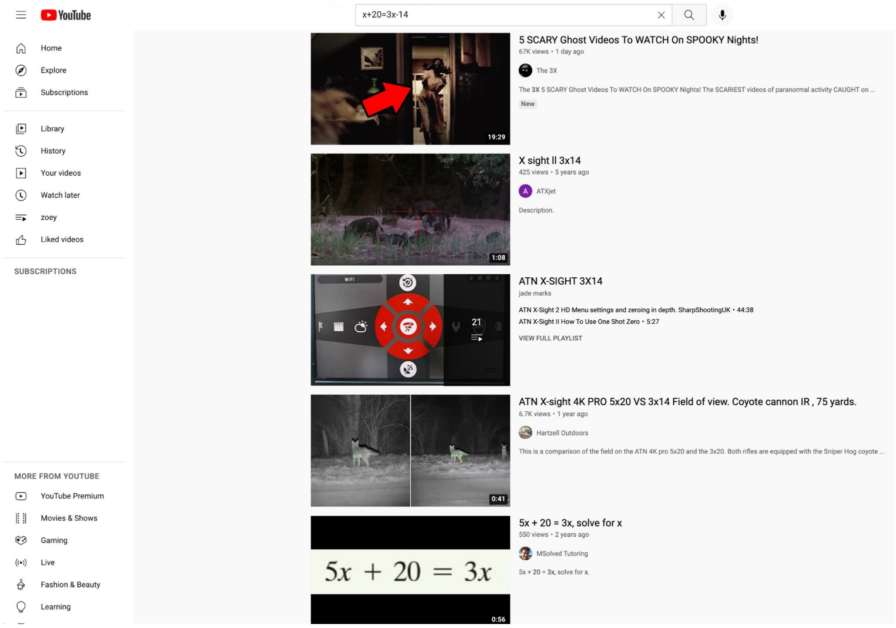
Image and audio of the video usually contain more detailed information of the content than title and description, especially in tutorial videos. The instructors always write down the problems related words and always explain how to solve the problems. Using image-extracted content information to match the search keywords might get more related tutorial video results. Thus, in this work, our goal is to explore if content-based information could be used to do video ranking on math tutorial videos. Specifically, we investigated how to use extracted math characters from images to represent a math video and use the representations to determine the similarities between an input query and a video. That is, based on a set of videos $V = \{v_1, v_2, \cdots, v_k\}$, we investigated how to build a representation $F = \{f_1, f_2, \cdots, f_k\}$ such that given an input math expression q_{in}, we could get a list of videos $V_r = [v_{r_1}, v_{r_2}, \cdots, v_{r_k}]$ sorted by the distance between the representation of q_{in} and the representation of v_{r_i} ($Dist(f(q_{in}), f_{r_i}$) in ascending order. $f_{r_i} = f(p(v_{r_i}))$, f is the representation map function, p is the pre-processing progress of the video, and $v_{r_i} \in V, i = 1, 2, \cdots, k$. Our main contribution in this project is that our proposed representation can beat the baseline and improve the quality of the videos searching results.

This chapter is structured as follows. First, we will discuss the previous work about content-based lecture search in Section 6.2. Second, we describe how we collected the dataset, the distance metrics and ranking accuracy metric that we used. Next, the different types of representations will be presented in Sections 6.4 to 6.6. Finally, the experiments we did and the results will be discussed in Sections 6.7 and 6.8.

6.2 Related Work

Due to the dramatic growth in the amount of the e-learning materials, many researchers have done a lot of work [21] on content-based lecture search. Since videos contain a lot of information in different channels (visual and audio), different researchers use them in different ways.

Visual: [115, 108, 1] only used text information that was extracted by Optical Character Recognition (OCR) models. [115] analyzed university classroom lecture videos that had slides. They used the detected text in slides to do topical event detection and synchronize the video with external documents (slides). In [108], the authors used OCR tools to detect keywords in a video segment and saved them to a database for achieving the "searching" function in their ICS (Indexing, Captioning, Searching) video framework on classroom videos. [1] applied a face detector to separate the slides frame and the speaker frame. They built a search engine on lecture videos with slides by using OCR tools to extract the



x+20=3x-14

5 SCARY Ghost Videos To WATCH On SPOOKY Nights!
67K views • 1 day ago
The 3X
The 3X 5 SCARY Ghost Videos To WATCH On SPOOKY Nights! The SCARIEST videos of paranormal activity CAUGHT on ...
New

X sight ll 3x14
425 views • 5 years ago
ATXjet
Description.

ATN X-SIGHT 3X14
jade marks
ATN X-Sight 2 HD Menu settings and zeroing in depth. SharpShootingUK • 44:38
ATN X-Sight II How To Use One Shot Zero • 5:27
VIEW FULL PLAYLIST

ATN X-sight 4K PRO 5x20 VS 3x14 Field of view. Coyote cannon IR , 75 yards.
6.7K views • 1 year ago
Hartzell Outdoors
This is a comparison of the field on the ATN 4K pro 5x20 and the 3x20. Both rifles are equipped with the Sniper Hog coyote ...

5x + 20 = 3x, solve for x
550 views • 2 years ago
MSolved Tutoring
5x + 20 = 3x, solve for x.

Figure 6.1: Example of searching a math problem in YouTube. In the returned top 5 videos, only 1 is a math tutorial video and it is not the one that the query asks.

text for indexing.

Audio: Audio channel information is used in [91, 68]. [91] used speech transcripts to build a chain index for browsing inside a video in a computer science course lecture video dataset. [68] extracted a text representation from speech transcripts and then clustered and ranked videos based on this representation in math tutorial videos dataset collected from YouTube.

Visual and Audio: Some other works [52, 10, 128, 127] combine the information from both visual channel and audio channel on lecture videos with slides. [52] did several experiments to show that the recall of the search result was tremendously improved when combined with the slides-based information and the audio-based information on lecture videos with slides while the precision was decreased. In [10], they summarized the lecture content by the content descriptive metadata by automatic key phrase extraction and topic-based segmentation using the detected slides text and the audio features. [128] built a navigation system by using the speech transcripts and the detected text from videos to do semantic video segmentation, and generating the table of content by analyzing the structure of the slide shots. In [127], the authors used both slide text and speech transcripts to extract video-level and segment-level keywords to achieve the lecture video indexing and browsing.

Our Work: In our work, we mainly explored if the representation extracted from the visual channel can be used in ranking math tutorial videos. This representation also could be used in searching content in a video if we stored the key frame representations into a database. The biggest difference between our work and others work is that their input query must exactly match the detected key works, whereas we do not require this.

6.3 Overview

Our goal is explore whether the extracted representation from images could be used to rank the tutorial videos. To be more specific, our research question is: how accurately can we rank videos using computer vision-based methods to match the input query about the solved math problem?

To achieve our goal, first we need to have a (1) dataset that contains math tutorial videos to work on. Second, there are a lot of information can be extracted from the image, and we need to explore what types of image (2) representations can be used and how to build the representations. Third, we need to decide which (3) distance metric should be used when we compare between different representations. After we have these, we need to design some (4) experiments to get the answer of our research question. The following sections will talk about these important parts in order.

6.4 Dataset: *Algebra*

We collected 242 math tutorial videos which have transcripts from YouTube using the keyword ”algebra” by YouTube Data API. For the reason that everyone can upload their own tutorial videos to YouTube, the videos in *Algebra* dataset by different authors have different teaching styles. Some authors use slides, while others prefer writing the problem solving process on whiteboard or paper. Some videos only contain the tutorial space (e.g., whiteboard, slides or paper), while others also contain tutor and a noise background.

Most of the videos presented multiple math problems. We submitted the collected videos to Amazon Mechanical Turk and asked the workers to watch and label all the solved problems in the unified format: *mm:ss;problem*. One video could have multiple labels. *mm:ss* is that time in the video that author started to talk about the *problem*. For example, the left image of Figure 6.2 is one frame in a video which is labelled as $00:58; y = 2x + 3$. It means at the problem $y = 2x + 3$ was started at 0 minutes 58 seconds in this video. The right is labelled as $07:31; 2(y - 8) = 24$. There are multiple problems in the slides, but at the time 7 minutes 31 seconds, only one problem, $2(y - 8) = 24$, is being solved and explained. Figure 6.4 shows an example of the Amazon Rekognition detection results.

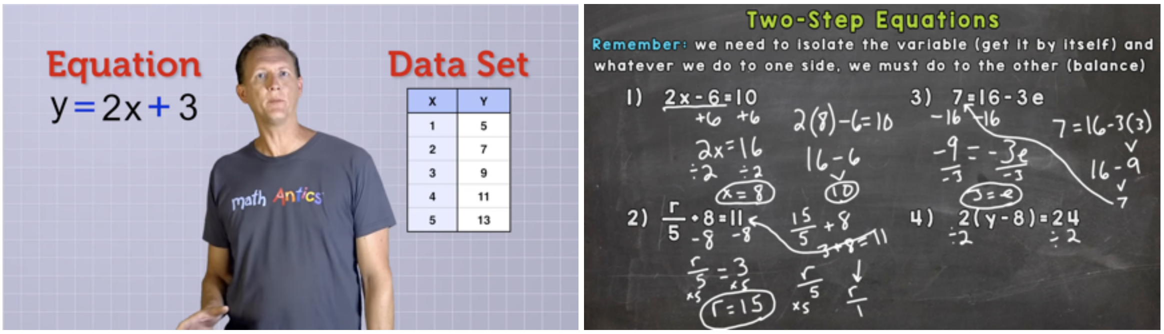

Equation
y=2x+3
Data Set
X Y
1 5
2 7
3 9
4 11
5 13
math Antics
Two-Step Equations
Remember: we need to isolate the variable (get it by itself) and whatever we do to one side, we must do to the other (balance)
1) 2x-6=10
2) r/5+8=11
3) 7=16-3e
4) 2(y-8)=24

Figure 6.2: Examples of Algebra dataset. **Left**: labelled as $y = 2x+3$; **Right**: labelled as $2(y-8) = 24$

We cropped the original videos according to the start time of the labels into short videos such that in each short video, it only contains one problem.

6.4.1 Data Cleaning

We did data cleaning on the dataset:

Label: First, we removed English words in the content of the problem label and only kept the math expressions. For example, we changed *0:00;Find the greatest common factor of these monomials: 10cd^2,25c^3d^2* to *0:00;10cd^2,25c^3d^2*. Second, we manually corrected some errors if the math expression was wrong.

Video: Third, we removed the videos in which there were no math expressions in the label. For example, *6:41;P(0,0,1)*.

After data cleaning, we obtained 1027 short videos in total. Each short video only has one problem label. For example, the new label for the left image in Figure 6.2 is $y = 2x + 3$ and the right one is $2(y - 8) = 24$.

The shortest time interval between two labels in the same video was 7 seconds. Thus, when we cropped the videos based on the start time in the labels, we assumed each problem was talked at least 5 seconds. We extracted 5 frames (1fps) for each short video.

Data details after cleaning: for each problem in the *Algebra* dataset, it has transcripts, images (5 frames) and the label that only contains math expression.

6.4.2 Training Dataset and Test Dataset

In the 1027 short videos, 1010 problems are unique and 15 are repeated. Two problems are repeated three times and thirteen problems are repeated twice. We split them randomly into training (687) and test (340) datasets, and they do not have any overlapped problems.

We compared several parameter-free representations of the video and picked the best one on the training dataset. Even though there were no parameters that need to be trained, we still used the training dataset results as a reference. Splitting the dataset can prevent data leakage.

6.5 Content-based Representation

Many tutors will write down important steps when they are explaining problems. These steps may include the statement problems, the key rules, and the solutions. For solving the same problem, different tutors will also write down similar content. This shared content can be a good representation for that problem. For different problems, the similarity between their representations should be less than it between the same problem.

We explored several different types of representation. They are Transcripts Representation, Feature Map Representation, Character-Vector Representation and Character-String Representation. Since we already have the transcripts in the dataset, and do not need extra steps to get it, we use the Transcripts Representation as our baseline.

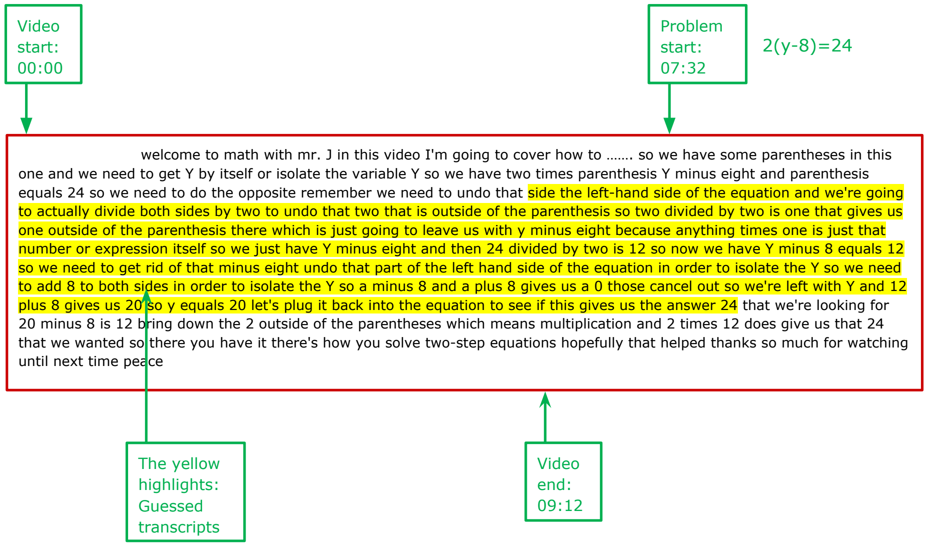
6.5.1 Transcripts Representation

The transcripts in the dataset is provided by Automatic Speech Recognition (ASR) model. It is not the ground truth transcripts for the video. It could have some errors.

The transcripts we get from YouTube is for the original long video. We don't have a timeline for the transcripts that can be used to extract the exact accurate transcripts for each short video. We thus used a segment of the entire transcripts as the *guessed transcript* for that problem. The segment is picked by the position of the problem in the original video. The unit is a word not a single character. Please see Fig. 6.3 as an example.

Pre-processing: We did pre-processing on the guessed transcripts. First, we only kept the math characters that we were interested in: (1) numbers ($0, 1, \cdots, 9$), (2) letters ($a, b, \cdots, z$) and (3) special characters: $\{+, -, =, (,), >, <, ?, /, *\}$. Next, some English words that can represent numbers and our interested special characters were changed into characters and kept. For example: from "three", "thirty-five", "subtract", "times" to $3, 35, -, *$. Third, some strings that were not related to math expression but appeared multiple times in the training dataset were also removed. For example, "nt", "etc","yay","hmm", "I".

Representation: We used a string as the Transcripts Representation of the problem in the video. After the pre-processing, we concatenated all the remaining characters in the guessed transcripts together by the original order.



welcome to math with mr. J in this video I'm going to cover how to .... so we have some parentheses in this one and we need to get Y by itself or isolate the variable Y so we have two times parenthesis Y minus eight and parenthesis equals 24 so we need to do the opposite remember we need to undo that side the left-hand side of the equation and we're going to actually divide both sides by two to undo that two that is outside of the parenthesis so two divided by two is one that gives us one outside of the parenthesis there which is just going to leave us with y minus eight because anything times one is just that number or expression itself so we just have Y minus eight and then 24 divided by two is 12 so now we have Y minus 8 equals 12 so we need to get rid of that minus eight undo that part of the left hand side of the equation in order to isolate the Y so we need to add 8 to both sides in order to isolate the Y so a minus 8 and a plus 8 gives us a 0 those cancel out so we're left with Y and 12 plus 8 gives us 20 so y equals 20 let's plug it back into the equation to see if this gives us the answer 24 that we're looking for 20 minus 8 is 12 bring down the 2 outside of the parentheses which means multiplication and 2 times 12 does give us that 24 that we wanted so there you have it there's how you solve two-step equations hopefully that helped thanks so much for watching until next time peace

Figure 6.3: Example of guessed transcripts

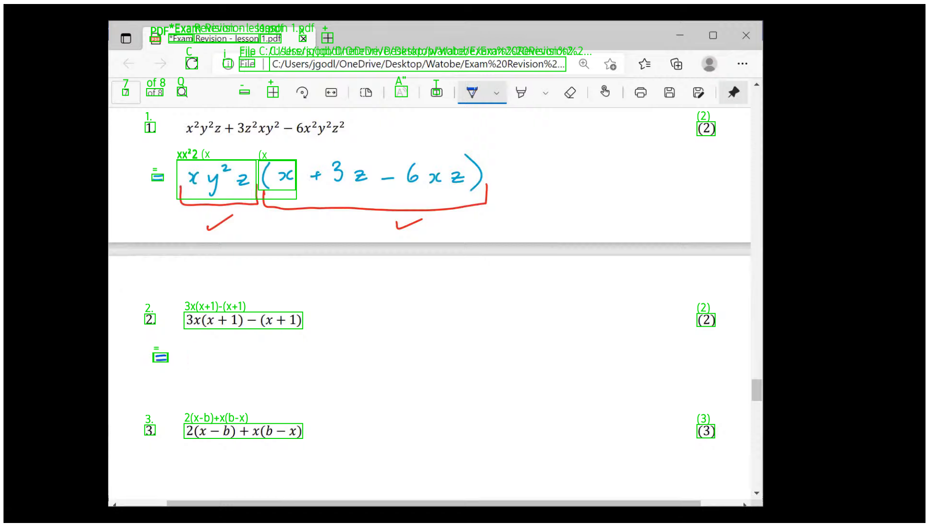
6.5.2 Feature Map Representation

Deep neural networks that are trained on ImageNet perform very well on many computer vision tasks such as image recognition [49], and object detection [89]. The layers in pre-trained ImageNet models already learned some useful features to represent objects in the images. This is the reason why people keep the initial layers of the pre-trained models, replace the last several layers and fine tune the new layers on their own tasks. In our case, the math content in the image might be already held in the feature maps. Therefore, it is an obvious approach to compare the feature maps of pre-trained model between problems in our dataset. We used cosine similarity to compare the feature maps.

6.5.3 Character-Vector Representation

Math problem detection is a special kind of text detection task. We can use a text detector to extract the math characters and use the detected information to represent the math problem in the image. Our work are based on analyzing the detection results of the Amazon Rekognition API. For each input image, the text detector of Amazon Rekognition API can provide the detected characters, locations (coordinates) and detection confidence scores. The detected characters could be either a single character or a string. If a string is detected, we will only have the coordinates for the whole string, not for each single character in this string. Figure 6.4 shows an example of Amazon Rekognition API detection results. We use cosine similarity to compare the vector representations.

Pre-processing: The text detector of Amazon Rekognition API will return all characters



1. $x^2y^2z + 3z^2xy^2 - 6x^2y^2z^2$ (2)

$xy^2z(x + 3z - 6xz)$

2. $3x(x+1) - (x+1)$ (2)

3. $2(x-b) + x(b-x)$ (3)

Exam Revision - lesson 1.pdf
C:/Users/jgodl/OneDrive/Desktop/Watobe/Exam%20Revision%2...
File
7
of 8
xx^2 (x
(x
3x(x+1)-(x+1)
2(x-b)+x(b-x)

Figure 6.4: Example of Amazon Rekognition API detection results

that can be detected in the image. Please see Fig. 6.4. The detection results have a lot of noise that we don't need for representing the math. We need to do pre-processing on it to retain only the math information. Similar to the pre-processing of guessed transcripts, first we only keep our interested math characters: (1) numbers, (2) letters, and (3) special characters: $\{+, -, =, (,), >, <, ?, /, *\}$. The total number of characters is 45. Second, We remove the detected sub-strings if they are in English dictionary with or without processed by the Porter Stemming Algorithm.

Representation: Three different vector representations are considered to represent the solved problems. (1) **1D Bag of Words** (1D BOW): It is a (1,45) vector that contains the number of occurrences for each interested character in that image. (2) **2D Bag of Words** (2D BOW): It is a (46, 45) matrix. The order of characters is important in math expression. For example, $3x + 2$ is different from $2x + 3$ even though they have the same 1D BOW representation. This 2D BOW representation considers the order of two characters into it. The top (45, 45) matrix contains the number of occurrences that each pair of characters appears immediately adjacent in the image. It is not a symmetric matrix. For example, if ”41” is detected once in the image, the element at [4,1] of top square (45,45) matrix will be 1; if ”14” is detected once, the element at [1,4] will be 1. The bottom (1,45) vector is the 1D BOW of the same frame. (3) **2D Confidence Map** (2D CM): This representation is similar to 2D BOW; it is also a (46, 45) matrix. Instead of the number of occurrences, the top (45,45) square matrix of 2D CM contains the how confident that two characters are detected together by the model in the image. The bottom (1, 45) vector is that how confident a single character is detected in the image. Sometimes the background of the

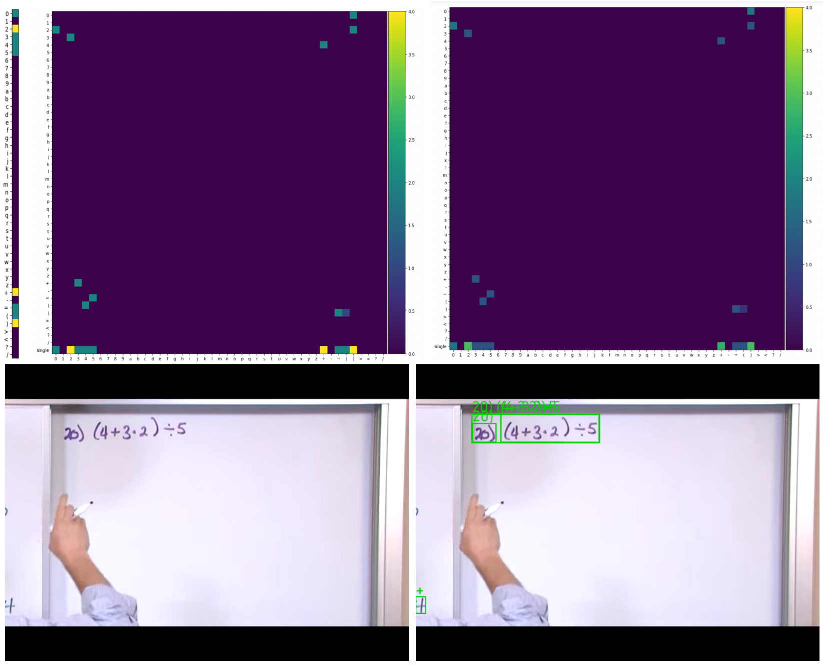


Figure 6.5: Top Left: 1D BOW. **Top Middle**: 2D BOW. **Top Right**: 2D CM. **Bottom Left**: The original image. **Bottom Right**: The detection results of Amazon Rekognition API. The three representations share the same color code.

video image is complex. The text detector could make some mistakes on the unclear pixels with a lower confidence score. This 2D CM might help in this situation. Figure 6.5 is an example of the 3 vector representations.

6.5.4 Character-String Representation

In this type of representation, we still use the detection results of Amazon Rekognition API and apply the same pre-processing progress on the detection results as in the Character-Vector Representation.

Representation: Here we use a string to represent the problem in the video. After the pre-processing, the remaining detected characters are concatenated together to be one string either by location or by the confidence score in the descending order. For example, if concatenated by location, the string representation for the same problem in Fig. 6.5 is $20)(4 + 3.2) = 5 + 20)(4 + 3.2) = 5+$; if by confidence score, it is $20)(4 + 3.2) = 5 + 20)(4 + 3.2) = 5+$.

6.6 Distance Metric and Accuracy Metric

For the feature map representation and character-vector representation, we used cosine similarity as the distance metric. For transcripts representation and character-string representation, we used edit distance and n-gram IOU as the distance metric to compare how similar of two problems.

6.6.1 String Distance Metric

Edit Distance: Edit distance is a method to measure the dissimilarity between two strings (a, b). It counts the number of operations required from one to another. There are three operations in edit distance: insertion, deletion and substitution.

Algorithm 1: Edit Distance

```
$n \leftarrow len(a)$;
$m \leftarrow len(b)$;
$distance[i,0] \leftarrow i \forall i \in 0,...,n$;
$distance[0,j] \leftarrow j \forall j \in 0,...,m$;
for $j \leftarrow 1$ to $m$ do
    for $i \leftarrow 1$ to $n$ do
        $indicator \leftarrow a[i] \neq b[j] ? 1 : 0$;
        $distance[i,j] \leftarrow$ Minimum ( $distance[i-1,j]+1$, $distance[i,j-1]+1$,
          $distance[i-1,j-1]+indicator$ );
    end
end
Return distance[n,m]
```

N-gram IOU: String IOU algorithm is similar to the object detection IOU. It is the length of intersection between two strings over the length of union of them.

However, to detect the math problem totally correct in handwritten is a very hard task. Here, we use a n-gram IOU instead of the original one. In this n-gram IOU algorithm, we calculate the IOU between each n length sub-string in the problem label (a) and the string representation (b), $n = 2, 3, ..., len(label)$. Since the shorter sub-string is easier to detect than the longer one, before adding all the sub-string IOU together, each sub-string IOU will be multiplied by a parameter α, $\alpha = \frac{n}{len(label)}$.

6.6.2 Ranking Measurement

We use Discounted Cumulative Gain (DCG) [112] to measure the quality of the returned video list. This measurement considers the order of the returned list. If there is no dupli-

Algorithm 2: N-gram IOU

```
$l \leftarrow len(a)$;
$m \leftarrow len(b)$;
$IOU\_score \leftarrow 0$;
for $n \leftarrow 2$ to $m$ do
    for $start\_idx \leftarrow 0$ to $m - n$ do
        $sub\_string = b[start\_idx : start\_idx + n]$;
        $\alpha = \frac{n}{l}$;
        $sub\_iou = \alpha * IOU(sub\_string, a)$ ;
        $IOU\_score = IOU\_score + sub\_iou$
    end
end
```

cate problem, the DCG of the best video list should be 1. If there are duplicate problems, the highest DCG can be greater than 1.

$$DCG_p = \sum_{i=1}^{p} \frac{x_i}{\log_2(i+1)}, \begin{cases} x_i = 1, \text{if the ith video is correct;} \\ x_i = 0, \text{if the ith video is wrong.} \end{cases}$$

For example, if the input query is $''x+2=5''$, output list A is $\{''x-2=5''$, $''x+2=5''$, $''x+2=3''\}$ and the output list B is $\{''x+2=5''$, $''x-2=5''$, $''x+2=3''\}$. The DCG of A is $0 + \frac{1}{log_2(3)} + \frac{0}{log_2(4)} = \frac{1}{log_2(3)} \approx 0.63$. The DCG of B is $1 + \frac{0}{log_2(3)} + \frac{0}{log_2(4)} = 1$. Thus, the output list B is better than the output list A.

6.7 Experiments

We applied the representations described above on the *Algebra* dataset. For each problem, we calculated the similarities between the label and all video representations in the dataset, returned a list of videos according to the similarity scores, and calculated the DCG score to measure how good it is. We compared different representations using the average DCG score over all the problems.

6.7.1 Sensitivity Analysis of String Representation

Besides the different type representations we talked about in the chapter 6.5, there are some potential factors that could influence the ranking results of string representation.

(1) **Time**: every instructor has their own teaching style. In some of the videos, the math expressions are there at the start of videos while in others, the tutors will write down

the solutions as time goes on. Thus, to use which frame's detection results to build the representation could be important.

(2) **Confidence Score**: The detection confidence score shows how likely the an object is in the image. A lower confidence score threshold means using more but less accurate math characters and a higher confidence score threshold means using less but more accurate math characters.

(3) **Special Characters**: Some of the relevant special characters represent the math relationship between two characters. For example, the character "*", "/". However, the detectors are usually very easy to make mistakes on them. Even though they contain some math information that may help to distinguish different problems, it is still not very clear that how much gain we can get by using them in the representation.

(4) **Distance Metric**: The returned list is ranked based on the similarity score. Different distance metrics will focus on different aspects of the compared contents. It is worth the effect to find a better one for the math representations.

(5) **String Concatenation Order**: The order of characters is very important in math problems. Different orders will create different problems. The importance of the order of the detected math objects depend on what are detected in them. For example, the order of "a", "b", "3", "$-$","$= 5$" is important than "$a - 3b$","$= 5$" for the problem "$a - 3b = 5$". The output order of the Amazon Rekognition API detection results is followed the geometric order: from the left to right, and top to down in the image. Concatenating the string by location can represent the math problem very well if there is only one problem in the image and no other unrelated text around the solving problem. However, in some videos, there are multiple problems in the same image. This might bring some confusion. Concatenating the text by the confidence score in descending order might reduce the influence by blurry or background noise.

6.8 Results and Analysis

The results of ImageNet-based feature map representation will not be stated in the same tables with others in the following analysis. It is much worse than the others. We used several feature maps on different layers of Resnet-18 and Resnet-50 which are pre-trained on ImageNet for object classification as the problem representation. The DCG score for the best one is 0.009. The average DCG of 10 random guess video list is 0.153. Object classification is a different task from math problem identification. The feature maps could contain the information that this image has some characters, but it is not enough for detecting what characters they are. Thus, for the rest of the work, we did not pursue deep learning-based methods but rather engineered our own methods based on the object (text)

Repr. Type	Train DCG	Test DCG
Transcripts	0.453	0.528
Vector	0.660	0.612
String	0.694	**0.715**

Table 6.1: Ranking results comparison between representation types

detection results to build the video representations.

The table 6.1 shows that our proposed string representation achieved the best ranking score (0.715) on the *Algebra* test dataset. Both the character-vector and character-string representation are better than the baseline. It could be the reason that the guessed transcripts are not very accurate. Moreover, the way to concatenate the characters may not be very appropriate for this representation. Compared between character-vector and character-string representation, the string representation contains more order information. For example, we can know if 3 or more characters appear together in the image.

6.8.1 Transcripts Representation

For the baseline, at first we applied pre-processing on the entire transcripts, the averaged DCG on training dataset is 0.304 and the averaged DCG on test dataset is 0.404. However, this is apparently not the best transcript for a problem. If the original video contains multiple problems, then these problems have the same transcripts. Using the guessed transcripts instead, the average DCG score on the training dataset is increased to 0.453 and the average DCG score on the test dataset is increased to 0.528.

This result shows that the transcripts has some information. An NLP model which is trained on the transcripts of math tutorial videos might help to output a better transcript-based representation.

6.8.2 Vector Representation

In this approach, the problem label is changed to a vector which is the same size as the vector representation. The 1D BOW of the label shows the number of occurrences of the characters in the label string; the 2D BOW contains the number of occurrences that a pair of characters are connected in the label string; the 2D CM is the same size as the 2D BOW.

The best one on *Algebra* of the vector representation is 2D Bag of Words; its average DCG on test dataset is 0.596 over the frame 1 and frame 5. Compared to 1D Bag of Words, it considers which pairs of characters appear together in the image. Apparently, this will help to distinguish different problems. 2D Bag of Words is one special case of 2D Confidence Map if we consider that every pair of characters has confidence score 1. The reason that 2D

Representation Name	Train DCG	Test DCG
$1D$ BOW	0.540	0.486
$2D$ BOW	0.650	**0.595**
$2D$ CM	0.632	0.584

Table 6.2: Average DCGs of different Vector representations over Frame 1 and Frame 5.

Representation Type	Frame #.	Train DCG	Test DCG
String	No.1	0.613	0.620
	No.5	0.639	**0.657**
Vector	No.1	0.559	0.509
	No.5	0.569	**0.539**

Table 6.3: Average DCG using detection math information from Frame 1 and Frame 5 of vector and string representation

BOW is slightly better could be that the Amazon Rekognition detection results are good. Even though the confidence score is lower, the detected character is correct.

6.8.3 String Representation

The best string representation achieved 0.715 DCG on the *Algebra* test dataset over all potential impact factors. Here is an example of the top 10 videos in the returned list by the representations with different DCGs. If the input math expression is 2st^2-s^2:

- 0.594 DCG: (s+t)^3, 2st^2-s^2, s/5+3/(5+2s/z), b-4, X^2+2X-15=0
- 0.715 DCG: 2st^2-s^2, (s+t)^3, 1/X^(-2), X^-4/1, (R-S)

Time: Selection of the key frame is a very important step. It decides what can be used to build the representation. From the table 6.3, both string representation and vector representation have better performance by the detection results on frame 5 (the frame on the 5th second). The results mean that there is more useful math information on the fifth frame than the first one. It could be the reason that the solved math problem is not in the image at the start of the video. And also it seems that most authors in our dataset would provide math solution step by step other than show everything at the beginning.

Confidence Score: As mentioned in previous chapter, this score indicates how confident the model is that the detected math characters are in the image. We applied the representation with different confidence score thresholds on frame 1, 3 and 5 of the short videos. The results in table 6.4 shows that the lower threshold is being used, the better the ranking performance is. A lower threshold will keep more math characters. This means that even the confidence score is low, the detection results are still relatively accurate.

Conf. Score	AVG Train DCG	AVG Test DCG
0	0.677	**0.692**
10	0.676	0.692
20	0.659	0.672
30	0.645	0.666
40	0.628	0.651
50	0.607	0.629
60	0.566	0.598
70	0.505	0.560
80	0.437	0.503
90	0.343	0.428

Table 6.4: Average string representation ranking results on frame 1, 3 and 5 with using different detection confidence score thresholds

With Special Characters	Train DCG	Test DCG
Yes	0.687	**0.707**
No	0.585	0.608

Table 6.5: String representation average DCG with or without special characters by 0 detection confidence score on frame 5

Special Characters: Similar to the transcripts representation, the special characters can improve the ranking results of string representation. Even though the detection results are not perfect, it still helps.

Distance Metric: An appropriate distance metric is important when comparing the similarity between two representations. In table 6.6, N-gram IOU outperforms the edit distance with a big advantage (0.715 vs 0.220).

Edit distance is a popular metric to measure how dissimilar between two strings. It is the minimum number of operations needed to change from one string to another among insertion, deletion and replacement. In our representation, the string could be either very long or very short. It depends on how many characters are detected. This will bring an issue that the algorithm will keep the label-length characters in the representation and replace them to be the same as the label. If two representations have similar length and the label-related characters is not in the beginning of the representations, these two will have similar large scores. We tried different penalty on deletion and replacement, the results are still not good.

The proposed N-gram IOU will give credits to all sub strings (length is greater than 2) in the representation if they are in the label. And the credits is depended on the length of the sub-string. It is more appropriate for this task.

String Concatenation Order: The plot 6.6 shows that the ranking results ordered by

Measurement Type	Edit Dist. Params	Train DCG	Test DCG
Edit Distance	Delete=1, Replace=1	0.199	0.220
	Delete=0.1, Replace=2	0.162	0.185
	Delete=0.01, Replace=2	0.160	0.182
N-gram IOU	-	0.694	**0.715**

Table 6.6: String representation ranking results with Edit distance using Frame 5. The *insert* penalty score is 1 for all cases.

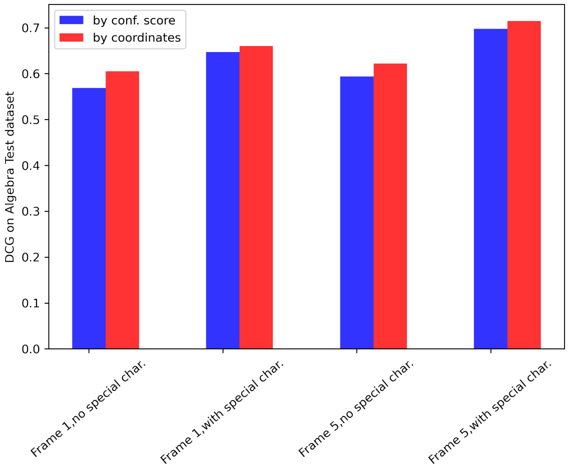


Figure 6.6: Concatenation Order impact

geometric information is better than those ordered by confidence score in each case on the *Algebra* test dataset.

6.9 Discussion and Conclusion

In this project, we collected a dataset from YouTube in which the authors were explaining single or multiple algebra problems. On the dataset, we explored whether the detected math characters could be used to rank the videos according to an input query (math expression). From our experiments, the proposed string representation is better than the vector and transcripts representation on our dataset. The more math information our representations included, the better the accuracy is: (1) the representation on Frame 5 is better than that on Frame 1; (2) the representation with special math characters is better than that without special math characters; (3) the representation using all detected characters is better than that using high detection confidence score characters. Also, the order of concatenating detected strings matters: (4) the representation that ordered by location is better than that ordered by detection confidence score. A suitable distance metric matters: (5) N-gram IOU is better than edit distance in our study.

For the students, the potential benefit is that they can thus find more accurate tutorial videos by using our system. In our dataset, all the videos are math related. Even though they are solving different problems, some common math characters can be used within them. For the dataset which contains math videos and other videos, the results could be even better, as the other videos might not contain any math characters in their key frames.

Future work: This project is an investigation on using the detected math characters to rank videos according the input query. There are some aspects that we can improve. (1) Build a math expression detector. The text detector of Amazon Recognition API is a regular text detector. It is good but not designed for math. It will make mistakes on some math expressions. (2) Design an algorithm to filter out the math unrelated characters in the detected results. Even though we used a package to find the variations of the English words and removed them, there were still some meaningful strings left. (3) For the baseline, a NLP model that can translate the sentence to math expressions might be useful. In our dataset, we also have the audio data. (4) We could apply the ASR model based on the start time of the problems to get a better guessed transcript.

Chapter 7

Conclusion

7.1 Summary

This dissertation presents my Ph.D. work about educational data mining with a long term goal to achieve personalized learning. The main work consists of two categories.

In the first category, I analyzed students' face videos to explore what can impact students' performance and emotions by computer vision-based methods in different learning settings. The most interesting and important finding in this category is that a new relationship between thermal comfort, learning and time is discovered. Thermal comfort may not significantly influence students' learning at the beginning, but later as time goes on, it can show stronger negative impacts on learning. In another work, the results show that the impact of short empathetic feedback messages of an ITS on students' emotions is very small and after considering the possible confounds, the impact may disappear entirely. I also developed a deep learning model to predict what is a good time for the human teachers to shift their eye gazes to the students during 1-on-1 tutoring session. It could be used to determine when to provide the attention to the students for personalized learning.

In the second part, I did work to improve educational content search by computer vision-based methods. The highlights in this category is that a new kind of dataset bias about the mathematical correctness of object configurations in the image is defined in the machine learning area. It has a small but consistent impact on the perception accuracy. In addition, I proposed new computer vision-based representations of videos to compare the similarities between math tutorial videos; this can enable more efficient search through educational content.

7.2 Directions for Future Work

In the educational data mining area, conducting experiments is a common method to investigate the factors that can influence students' learning. Based on my experience, it is better to provide more thorough instructions to the participants about how to finish the experiment instead of telling them the detailed goal of the experiments. A detailed goal might accidentally influence the participants' actions. For example, it could happen that some participants believe that they are extremely thermal comfortable in 25° while extremely uncomfortable in 30°. It could be the reason that they know that we want to explore the relationship between the room temperature and the thermal comfort. They want to help us distinguish the different thermal comfort feelings between the two different room temperatures. However, this cannot help us to find the true relationships between them.

My research has also suggested a perspective on future research on developing personalized learning systems: It may not always be better to follow students' preferences. Their preferences will inevitably be optimal for their feelings but might not be optimal for their learning. For example, some students might not want any attention from the teachers because they will feel nervous. However, these students could be the ones that need a lot of attentions to keep engaged in the course.

Finally, in the educational data mining area, a better object detector which is designed for math characters is in demand. Most of the text detectors (e.g., Amazon Rekognition) are focused on detecting regular text detectors. They will miss a lot of important math information from images. A math character detector could be very useful in homework grading and educational video memorization, indexing and searching.

Bibliography

[1] J. Adcock, M. Cooper, L. Denoue, H. Pirsiavash, and L. A. Rowe. Talkminer: a lecture webcast search engine. In *Proceedings of the 18th ACM international conference on Multimedia*, pages 241–250, 2010.

[2] A. Agrawal, D. Batra, and D. Parikh. Analyzing the behavior of visual question answering models. *arXiv preprint arXiv:1606.07356*, 2016.

[3] F. Ahmed, Y. Bengio, H. van Seijen, and A. Courville. Systematic generalisation with group invari-ant predictions. 2021.

[4] Y. Al Horr, M. Arif, A. Kaushik, A. Mazroei, M. Katafygiotou, and E. Elsarrag. Occupant productivity and office indoor environment quality: A review of the literature. *Building and environment*, 105:369–389, 2016.

[5] T. C. S. Andallaza and R. J. M. Jimenez. Design of an affective agent for aplusix. *Undergraduate thesis, Ateneo de Manila University, Quezon City*, 2012.

[6] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. L. Zitnick, and D. Parikh. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pages 2425–2433, 2015.

[7] M. Arif, M. Katafygiotou, A. Mazroei, A. Kaushik, E. Elsarrag, et al. Impact of indoor environmental quality on occupant well-being and comfort: A review of the literature. *International Journal of Sustainable Built Environment*, 5(1):1–11, 2016.

[8] I. Arroyo, B. P. Woolf, D. G. Cooper, W. Burleson, and K. Muldner. The impact of animated pedagogical agents on girls' and boys' emotions, attitudes, behaviors and learning. In *International Conference on Advanced Learning Technologies*, 2011.

[9] ASHRAE. Standard 55-2004. thermal environmental conditions for human occupancy. *American Society of Heating, Refrigerating and Air-Conditioning Engineers*, 2004.

[10] V. Balasubramanian, S. G. Doraisamy, and N. K. Kanakarajan. A multimodal approach for extracting content descriptive metadata from lecture videos. *Journal of Intelligent Information Systems*, 46(1):121–145, 2016.

[11] T. Baltrušaitis, A. Zadeh, Y. Chong Lim, and L.-P. Morency. Openface 2.0: Facial behavior analysis toolkit. 2018.

[12] P. Barrett, F. Davies, Y. Zhang, and L. Barrett. The impact of classroom design on pupils' learning: Final results of a holistic, multi-level analysis. *Building and Environment*, 89:118–133, 2015.

[13] P. Barrett, Y. Zhang, J. Moffat, and K. Kobbacy. A holistic, multi-level analysis identifying the impact of classroom design on pupils' learning. *Building and environment*, 59:678–689, 2013.

[14] C. Bidet-Ildei, M. Gimenes, L. Toussaint, Y. Almecija, and A. Badets. Sentence plausibility influences the link between action words and the perception of biological human movements. *Psychological research*, 81(4):806–813, 2017.

[15] A. Borji and L. Itti. State-of-the-art in visual attention modeling. *IEEE transactions on pattern analysis and machine intelligence*, 35(1):185–207, 2013.

[16] A. Borji, D. N. Sihite, and L. Itti. What/where to look next? modeling top-down visual attention in complex interactive environments. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 44(5):523–538, 2014.

[17] A. Borji, H. R. Tavakoli, D. N. Sihite, and L. Itti. Analysis of scores, datasets, and models in visual saliency prediction. In *Proceedings of the IEEE international conference on computer vision*, pages 921–928, 2013.

[18] N. Bosch, S. K. D'mello, J. Ocumpaugh, R. S. Baker, and V. Shute. Using video to automatically detect learner affect in computer-enabled classrooms. *ACM Transactions on Interactive Intelligent Systems (TiiS)*, 6(2):17, 2016.

[19] G. S. Brager and R. J. De Dear. Thermal adaptation in the built environment: a literature review. *Energy and buildings*, 27(1):83–96, 1998.

[20] A. H. Buckman, M. Mayfield, and S. B. Beck. What is a smart building? *Smart and Sustainable Built Environment*, 2014.

[21] D. Chand and H. Ogul. Content-based search in lecture video: A systematic literature review. In *2020 3rd International Conference on Information and Computer Technologies (ICICT)*, pages 169–176. IEEE, 2020.

[22] S. Cho, J. Lim, C. Hickey, and B.-T. Zhang. Problem difficulty in arithmetic cognition: Humans and connectionist models. 2019.

[23] S. Choi, D. A. Guerin, H.-Y. Kim, J. K. Brigham, and T. Bauer. Indoor environmental quality of classrooms and student outcomes: A path analysis approach. *Journal of Learning Spaces*, 2(2):2013–2014, 2014.

[24] S. Claro, D. Paunesku, and C. S. Dweck. Growth mindset tempers the effects of poverty on academic achievement. *Proceedings of the National Academy of Sciences*, 113(31):8664–8668, 2016.

[25] R. De Dear and G. S. Brager. Developing an adaptive model of thermal comfort and preference. 1998.

[26] Ö. De Manzano, T. Theorell, L. Harmat, and F. Ullén. The psychophysiology of flow during piano playing. *Emotion*, 10(3):301, 2010.

[27] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[28] S. D'Mello, T. Jackson, S. Craig, B. Morgan, P. Chipman, H. White, N. Person, B. Kort, R. el Kaliouby, R. Picard, et al. Autotutor detects and responds to learners affective and cognitive states. In *Proc. Emotional and Cognitive Issues Workshop at Int. Conf. Intelligent Tutoring Systems*, 2008.

[29] S. M. Doane and Y. W. Sohn. Adapt: A predictive cognitive model of user visual attention and action planning. *User Modeling and User-Adapted Interaction*, 10(1):1–45, 2000.

[30] P. V. Dorizas, M.-N. Assimakopoulos, and M. Santamouris. A holistic approach for the assessment of the indoor environmental quality, student productivity, and energy consumption in primary schools. *Environmental monitoring and assessment*, 187(5):259, 2015.

[31] K. Dwivedi, K. Biswaranjan, and A. Sethi. Drowsy driver detection using representation learning. In *International advanced computing conference (IACC)*, 2014.

[32] K. Dykstra, J. Whitehill, L. Salamanca, M. Lee, A. Carini, J. Reilly, and M. Bartlett. Modeling one-on-one tutoring sessions. In *2012 Proc. IEEE Int. Conf. Development and Learning and Epigenetic Robotics*, pages 1–2. IEEE, 2012.

[33] S. D'Mello, B. Lehman, J. Sullins, R. Daigle, R. Combs, K. Vogt, L. Perkins, and A. Graesser. A time for emoting: When affect-sensitivity is and isn't effective at promoting deep learning. In *Intelligent tutoring systems*, 2010.

[34] R. Ekman. *What the face reveals: Basic and applied studies of spontaneous expression using the Facial Action Coding System (FACS)*. Oxford UP, USA, 1997.

[35] P. Fanger. Moderate thermal environments determination of the pmv and ppd indices and specification of the conditions for thermal comfort. *ISO 7730*, 1984.

[36] P. O. Fanger et al. Thermal comfort. analysis and applications in environmental engineering. *Thermal comfort. Analysis and applications in environmental engineering.*, 1970.

[37] M. Feidakis, T. Daradoumis, and S. Caballé. Emotion measurement in intelligent tutoring systems: what, when and how to measure. In *International Conference on Intelligent Networking and Collaborative Systems*, 2011.

[38] S. Feng, J. Stewart, D. Clewley, and A. C. Graesser. Emotional, epistemic, and neutral feedback in autotutor trialogues to improve reading comprehension. In *International Conference on Artificial Intelligence in Education*. Springer, 2015.

[39] S. Frintrop, E. Rome, and H. I. Christensen. Computational visual attention systems and their cognitive foundations: A survey. *ACM Transactions on Applied Perception (TAP)*, 7(1):6, 2010.

[40] R. Fry and G. F. Smith. The effects of feedback and eye contact on performance of a digit-coding task. *The Journal of Social Psychology*, 96(1):145–146, 1975.

[41] R. Geirhos, P. Rubisch, C. Michaelis, M. Bethge, F. A. Wichmann, and W. Brendel. Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. *arXiv preprint arXiv:1811.12231*, 2018.

[42] A. Gilavand. Investigating the impact of environmental factors on learning and academic achievement of elementary students. *Health Sciences*, 5(7S):360–369, 2016.

[43] S. Gould, J. Arfvidsson, A. Kaehler, B. Sapp, M. Messner, G. R. Bradski, P. Baumstarck, S. Chung, A. Y. Ng, et al. Peripheral-foveal vision for real-time object recognition and tracking in video. In *IJCAI*, volume 7, pages 2115–2121, 2007.

[44] A. Graesser, B. McDaniel, P. Chipman, A. Witherspoon, S. D'Mello, and B. Gholson. Detection of emotions during learning with autotutor. In *Proceedings of the 28th annual meetings of the cognitive science society*, pages 285–290. Citeseer, 2006.

[45] A. C. Graesser and S. K. D'mello. Affect-sensitive intelligent tutoring system, Aug. 20 2019. US Patent 10,388,178.

[46] E. Gu and N. Badler. Visual attention and eye gaze during multiparty conversations with distractions. In *Intelligent Virtual Agents*, pages 193–204. Springer, 2006.

[47] Q. Guo, Y. Qian, and X. Liang. Mining logic patterns from visual data. In *2019 International Conference on Data Mining Workshops (ICDMW)*, pages 620–627. IEEE Computer Society, 2019.

[48] U. Haverinen-Shaughnessy, D. Moschandreas, and R. Shaughnessy. Association between substandard classroom ventilation rates and students' academic achievement. *Indoor air*, 21(2):121–131, 2011.

[49] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[50] O. W. Hill, Z. Serpell, and M. O. Faison. The efficacy of the learningrx cognitive training program: modality and transfer effects. *The Journal of Experimental Education*, 84(3):600–620, 2016.

[51] Y. Hoshen and S. Peleg. Visual learning of arithmetic operation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 30, 2016.

[52] W. Hürst, T. Kreuzer, and M. Wiesenhütter. A qualitative study towards using large vocabulary automatic speech recognition to index recorded presentations for search and access over the web. In *ICWI*, pages 135–143. Citeseer, 2002.

[53] F. Jazizadeh and W. Jung. Personalized thermal comfort inference using rgb video images for distributed hvac control. *Applied Energy*, 220:829–841, 2018.

[54] H. Jiang, K. Dykstra, and J. Whitehill. Predicting when teachers look at their students in 1-on-1 tutoring sessions. In *2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)*, pages 593–598. IEEE, 2018.

[55] H. Jiang, M. Iandoli, S. Van Dessel, S. Liu, and J. Whitehill. Measuring students' thermal comfort and its impact on learning. *Educational Data Mining*, 2019.

[56] H. Jiang and E. Learned-Miller. Face detection with the faster r-cnn. In *Automatic Face & Gesture Recognition (FG 2017), 2017 12th IEEE International Conference on*, pages 650–657. IEEE, 2017.

[57] J. Jiang, D. Wang, Y. Liu, Y. Xu, and J. Liu. A study on pupils' learning performance and thermal comfort of primary schools in china. *Building and Environment*, 134:102–113, 2018.

[58] J. Johnson, B. Hariharan, L. Van Der Maaten, L. Fei-Fei, C. Lawrence Zitnick, and R. Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2901–2910, 2017.

[59] A. Joshi, D. Allessio, J. Magee, J. Whitehill, I. Arroyo, B. Woolf, S. Sclaroff, and M. Betke. Affect-driven learning outcomes prediction in intelligent tutoring systems. In *Automatic Face & Gesture Recognition*, 2019.

[60] W. Jung and F. Jazizadeh. Vision-based thermal comfort quantification for hvac control. *Building and Environment*, 2018.

[61] S. Kaltwang, O. Rudovic, and M. Pantic. Continuous pain intensity estimation from facial expressions. In *International Symposium on Visual Computing*, pages 368–377. Springer, 2012.

[62] K.-i. Kameda, S. Murakami, K. Ito, and T. Kaneko. Study on productivity in the classroom (part 3) nationwide questionnaire survey on the effects of ieq on learning. *Clima 2007 WellBeing Indoors*, 2006(Part 3), 2007.

[63] S. Karumbaiah, R. Lizarralde, D. Allessio, B. Woolf, I. Arroyo, and N. Wixon. Addressing student behavior and affect with empathy and growth mindset. *Educational Data Mining*, 2017.

[64] B. Kort and R. Reilly. An affective module for an intelligent tutoring system. In *Intelligent Tutoring Systems*, pages 955–962. Springer, 2002.

[65] B. Kort, R. Reilly, and R. W. Picard. An affective model of interplay between emotions and learning: Reengineering educational pedagogy-building a learning companion. In *2001 Proc. IEEE Int. Conf. Advanced Learning Technologies*, pages 43–46, 2001.

[66] A. S. Lan, A. Botelho, S. Karumbaiah, R. S. Baker, and N. Heffernan. Accurate and interpretable sensor-free affect detectors via monotonic neural networks. In *International Conference on Learning Analytics & Knowledge*, 2020.

[67] M. Lee, K. Mui, L. Wong, W. Chan, E. Lee, and C. Cheung. Student learning performance and indoor environmental quality (ieq) in air-conditioned university teaching rooms. *Building and Environment*, 49:238–244, 2012.

[68] P. Liamthong and J. Whitehill. *Text Representations of Math Tutorial Videos for Clustering, Retrieval, and Learning Gain Prediction*. PhD thesis, WORCESTER POLYTECHNIC INSTITUTE, 2021.

[69] A. Lipczynska, S. Schiavon, and L. T. Graham. Thermal comfort and self-reported productivity in an office with ceiling fans in the tropics. *Building and Environment*, 135:202–212, 2018.

[70] S. Liu, S. Schiavon, A. Kabanshi, and W. W. Nazaroff. Predicted percentage dissatisfied with ankle draft. *Indoor air*, 27(4):852–862, 2017.

[71] S. Liu, Z. Zhang, K. Song, and B. Zeng. Arithmetic addition of two integers by deep image classification networks: experiments to quantify their autonomous reasoning ability. *arXiv preprint arXiv:1912.04518*, 2019.

[72] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016.

[73] W. Liu, Z. Lian, Q. Deng, and Y. Liu. Evaluation of calculation methods of mean skin temperature for use in thermal comfort study. *Building and Environment*, 46(2):478–488, 2011.

[74] G. C. Marchand, N. M. Nardi, D. Reynolds, and S. Pamoukov. The impact of the classroom built environment on student perceptions and learning. *Journal of Environmental Psychology*, 40:187–197, 2014.

[75] D. McNeill. *Hand and mind: What gestures reveal about thought*. University of Chicago Press, 1992.

[76] A. L. Mondragon, R. Nkambou, and P. Poirier. Evaluating the effectiveness of an affective tutoring agent in specialized education. In *European conference on technology enhanced learning*, pages 446–452. Springer, 2016.

[77] H. Nguyen and J. Masthoff. Designing empathic computers: the effect of multimodal empathic feedback using animated agent. In *Proceedings of the 4th international conference on persuasive technology*, pages 1–9, 2009.

[78] B. Nollet, M. Lefort, and F. Armetta. Learning arithmetic operations with a multi-step deep learning. In *2020 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2020.

[79] A. Oliva and A. Torralba. The role of context in object recognition. *Trends in cognitive sciences*, 11(12):520–527, 2007.

[80] J. P. Otteson and C. R. Otteson. Effect of teacher's gaze on children's story recall. *Perceptual and Motor Skills*, 50(1):35–42, 1980.

[81] B. Pavlin, G. Pernigotto, F. Cappelletti, P. Bison, R. Vidoni, and A. Gasparella. Real-time monitoring of occupants' thermal comfort through infrared imaging: A preliminary study. *Buildings*, 7(1):10, 2017.

[82] C. I. Penaloza, Y. Mae, K. Ohara, and T. Arai. Using depth to increase robot visual attention accuracy during tutoring. In *IEEE International Conference on Humanoid Robots - Workshop of Developmental Robotics*, 2012.

[83] P. Pham and J. Wang. Attentivelearner: improving mobile mooc learning via implicit heart rate tracking. In *International conference on artificial intelligence in education*, pages 367–376. Springer, 2015.

[84] V. Ramanathan, J. Huang, S. Abu-El-Haija, A. Gorban, K. Murphy, and L. Fei-Fei. Detecting events and key actors in multi-person videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3043–3053, 2016.

[85] H. Ranjbartabar, D. Richards, A. Bilgin, and C. Kutay. First impressions count! the role of the human's emotional state on rapport established with an empathic versus neutral virtual therapist. *IEEE transactions on affective computing*, 2019.

[86] K. Rayner, T. Warren, B. J. Juhasz, and S. P. Liversedge. The effect of plausibility on eye movements in reading. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 30(6):1290, 2004.

[87] A. Recasens, A. Khosla, C. Vondrick, and A. Torralba. Where are they looking? In *Advances in Neural Information Processing Systems*, pages 199–207, 2015.

[88] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

[89] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015.

[90] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE transactions on pattern analysis and machine intelligence*, 39(6):1137–1149, 2017.

[91] S. Repp, A. Gross, and C. Meinel. Browsing within lecture videos based on the chain index of speech transcription. *IEEE Transactions on learning technologies*, 1(3):145–156, 2008.

[92] J. Robison, S. McQuiggan, and J. Lester. Evaluating the consequences of affective feedback in intelligent tutoring systems. In *Affective computing and intelligent interaction and workshops*, 2009.

[93] S. A. Samani and S. A. Samani. The impact of indoor lighting on students' learning performance in learning environments: A knowledge internalization perspective. *International Journal of Business and Social Science*, 3(24), 2012.

[94] I. Sarbu and C. Pacurar. Experimental and numerical research to assess indoor environment quality and schoolwork performance in university classrooms. *Building and Environment*, 93:141–154, 2015.

[95] A. Sarrafzadeh, H. G. Hosseini, C. Fan, and S. P. Overmyer. Facial expression analysis for estimating learner's emotional state in intelligent tutoring systems. In *International Conference on Advanced Technologies*, 2003.

[96] L. Saulter, K. Thomas, Y. Lin, J. Whitehill, and Z. Serpell. Detecting affect over four days of cognitive training. Poster presented at the Temporal Dynamics of Learning Center All-Hands Meeting at UCSD, 2013.

[97] R. Sawyer, A. Smith, J. Rowe, R. Azevedo, and J. Lester. Enhancing student models in game-based learning with facial expression recognition. In *User modeling, adaptation and personalization*, 2017.

[98] B. Schölkopf, F. Locatello, S. Bauer, N. R. Ke, N. Kalchbrenner, A. Goyal, and Y. Bengio. Toward causal representation learning. *Proceedings of the IEEE*, 109(5):612–634, 2021.

[99] O. Seppanen, W. J. Fisk, and Q. Lei. Room temperature and productivity in office work. 2006.

[100] O. A. Seppänen and W. Fisk. Some quantitative relations between indoor environmental quality and work performance or health. *Hvac&R Research*, 12(4):957–973, 2006.

[101] H. Shah, K. Tamuly, A. Raghunathan, P. Jain, and P. Netrapalli. The pitfalls of simplicity bias in neural networks. *arXiv preprint arXiv:2006.07710*, 2020.

[102] P. Shayan and M. van Zaanen. Predicting student performance from their behavior in learning management systems. *International Journal of Information and Education Technology*, 9(5):337–341, 2019.

[103] J. V. Sherwood. Facilitative effects of gaze upon learning. *Perceptual and Motor Skills*, 64(3c):1275–1278, 1987.

[104] N. Spolaor, H. D. Lee, W. S. R. Takaki, L. A. Ensina, C. S. R. Coy, and F. C. Wu. A systematic review on content-based video retrieval. *Engineering Applications of Artificial Intelligence*, 90:103557, 2020.

[105] N. Srinivasan. Progress in brain research: Attention. 2009.

[106] J. F. Thayer, F. Åhs, M. Fredrikson, J. J. Sollers III, and T. D. Wager. A meta-analysis of heart rate variability and neuroimaging studies: implications for heart rate variability as a marker of stress and health. *Neuroscience & Biobehavioral Reviews*, 36(2):747–756, 2012.

[107] V. Todea. Guide for psycho diagnosis laboratory. *Timisoara: Artpress Publishing House (in Romanian)*, 2008.

[108] T. Tuna, J. Subhlok, L. Barker, V. Varghese, O. Johnson, and S. Shah. Development and evaluation of indexed captioned searchable videos for stem coursework. In *Proceedings of the 43rd ACM technical symposium on Computer Science Education*, pages 129–134, 2012.

[109] S. Turkay and S. T. Moulton. The educational impact of whiteboard animations: An experiment using popular social science lessons. In *Proceedings of the 7th International Conference of Learning International Networks Consortium (LINC). Cambridge, MA, USA*, pages 283–91, 2016.

[110] N. van den Bogert, J. van Bruggen, D. Kostons, and W. Jochems. First steps into understanding teachers' visual perception of classroom events. *Teaching and Teacher Education*, 37:208–216, 2014.

[111] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 1, pages I–I. IEEE, 2001.

[112] E. M. Voorhees. Evaluation by highly relevant documents. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 74–82, 2001.

[113] E. Vural, M. Bartlett, G. Littlewort, M. Cetin, A. Ercil, and J. Movellan. Discrimination of moderate and acute drowsiness based on spontaneous facial expressions. In *2010 20th International Conference on Pattern Recognition*, pages 3874–3877. IEEE, 2010.

[114] D. Wang, H. Zhang, E. Arens, and C. Huizenga. Observations of upper-extremity skin temperature and corresponding overall-body thermal sensations and comfort. *Building and Environment*, 42(12):3933–3943, 2007.

[115] F. Wang, C.-W. Ngo, and T.-C. Pong. Structuring low-quality videotaped lectures for cross-reference browsing by video text analysis. *Pattern Recognition*, 41(10):3257–3269, 2008.

[116] A. Wangperawong. Attending to mathematical language with transformers. *arXiv preprint arXiv:1812.02825*, 2018.

[117] P. Wargocki and D. P. Wyon. The effects of moderately raised classroom temperatures and classroom ventilation rate on the performance of schoolwork by children (rp-1257). *Hvac&R Research*, 13(2):193–220, 2007.

[118] J. Whitehill, M. Bartlett, and J. Movellan. Automatic facial expression recognition for intelligent tutoring systems. In *Computer Vision and Pattern Recognition Workshops, 2008. CVPRW'08. IEEE Computer Society Conference on*, pages 1–6. IEEE, 2008.

[119] J. Whitehill, Z. Serpell, Y.-C. Lin, A. Foster, and J. R. Movellan. The faces of engagement: Automatic recognition of student engagementfrom facial expressions. *Affective Computing, IEEE Transactions on*, 5(1):86–98, 2014.

[120] C. L. Widmer. *Examining the Impact of Dialogue Moves in Tutor-Learner Discourse Using a Wizard of Oz Technique*. PhD thesis, Miami University, 2017.

[121] B. Woolf, W. Burleson, I. Arroyo, T. Dragon, D. Cooper, and R. Picard. Affect-aware tutors: recognising and responding to student affect. *Int. Journal of Learning Technology*, 4(3):129–164, 2009.

[122] H.-Y. Wu, M. Rubinstein, E. Shih, J. Guttag, F. Durand, and W. Freeman. Eulerian video magnification for revealing subtle changes in the world. 2012.

[123] H. Xie, H.-C. Chu, G.-J. Hwang, and C.-C. Wang. Trends and development in technology-enhanced adaptive/personalized learning: A systematic review of journal publications from 2007 to 2017. *Computers & Education*, 140:103599, 2019.

[124] X. Xie, K. Siau, and F. F.-H. Nah. Covid-19 pandemic–online education in the new normal and the next normal. *Journal of information technology case and application research*, 22(3):175–187, 2020.

[125] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *International Conference on Machine Learning*, pages 2048–2057, 2015.

[126] Z. Yan and X. S. Zhou. How intelligent are convolutional neural networks? *arXiv preprint arXiv:1709.06126*, 2017.

[127] H. Yang and C. Meinel. Content based lecture video retrieval using speech and video text information. *IEEE Transactions on learning technologies*, 7(2):142–154, 2014.

[128] B. Zhao, S. Lin, X. Luo, S. Xu, and R. Wang. A novel system for visual navigation of educational videos using multimodal cues. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 1680–1688, 2017.

[129] Z. S. Zomorodian, M. Tahsildoost, and M. Hafezi. Thermal comfort in educational buildings: A review article. *Renewable and sustainable energy reviews*, 59:895–906, 2016.