Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence

A Hybrid Grammar-Based Approach for Learning and Recognizing Natural Hand Gestures

Amir Sadeghipour and **Stefan Kopp**
Faculty of Technology, Center of Excellence 'Cognitive Interaction Technology' (CITEC),
Bielefeld University, P.O. Box 100131, D-33501 Bielefeld, Germany

Abstract

In this paper, we present a hybrid grammar formalism designed to learn structured models of natural iconic gesture performances that allow for compressed representation and robust recognition. We analyze a dataset of iconic gestures and show how the proposed Feature-based Stochastic Context-Free Grammar (FSCFG) can generalize over both structural and feature-based variations among different gesture performances.

Introduction

Natural gestures are becoming increasingly popular as a means to interact with technical systems, from tablet devices, to smart TVs, to social robots. However, the gestures that can be used are severely restricted to what can be recognized robustly. Although there have been advances in vision-based motion tracking, these have been usually confined to predefined, reliably discernable movement patterns. Successes in this area include Hidden Markov Models (HMM) (Yamato, Ohya, and Ishii 1992; Yoon et al. 2001; Turaga et al. 2008), HMM in combination with the Kalman filter (Ramamoorthy et al. 2003), or the Ordered Means Model (OMM) (Großekathöfer et al. 2012). However, although these gestures are often reminiscent of familiar physical motions, they remain artificially defined gesture commands.

Natural gesturing, on the other hand, has for the most part resisted attempts at reliable recognition and interpretation. One reason is the wide variability or apparent lack of structural patterns that can be tapped by classifiers. For instance, iconic gestures are naturally performed during communication to refer to objects or events by depicting aspects of their visual-spatial properties. However, one can refer to a ball by drawing a circle with an index finger, with either hand or both hands simultaneously, clockwise or counter-clockwise, slow or fast, small or big, once or repeated several times. Hence, recognizing and interpreting an iconic gesture requires learning discriminative models that can recognize the different variants without overgeneralization. Furthermore, it needs to be robust against minor motion deviations or motion tracking noises.

A number of researchers have tried to develop linguistically motivated grammar-based approaches that can capture highly complex hierarchical structures, and recognize human actions and activities (Bobick and Ivanov 1998; Pastra and Aloimonos 2012). In this view, there is a grammatical structure behind nonverbal behavior that can be used to recognize or "parse" valid performances. This idea builds on the strong assumption that human actions and movements consists of primitives (like morphemes or words in language) that are arranged in compositional structures in a limited number of ways. However, when looking at gestures or, more generally, human motor actions, it is not clear what such primitives may be, whether they are stable, and what they are composed of. Hence, most grammar-based approaches for activity recognition have used a syntax that has been predefined by experts (Ivanov and Bobick 2000; Moore and Essa 2002; Chen, Georganas, and Petriu 2008).

In this paper, we present an approach to learning structured models of gesture performance that allows for a compressed representation and robust recognition of natural iconic gestures. We propose a *hybrid* approach that, at the same time, strives to extract the structural-syntactic patterns of gestures while identifying low-level statistical regularities that constitute primitive building blocks. After reviewing related work on grammar-based approaches used in computer vision and behavior recognition, we propose *Feature-based Stochastic Context-Free Grammars (FSCFG)* as a hybrid approach that extends Stochastic Context-Free Grammars (SCFG; proposed by Stolcke 1994) by using sets of probabilistically defined features as terminal symbols. The proposed hybrid grammar has been applied to learning and recognizing diverse natural iconic gestures from noisy movement data delivered by standard tracking sensors. Before reporting the classification results, we describe our recorded corpus comprising 1739 iconic gestures performed in reference to twenty different objects. We also analyze the strategies the participants used to depict an object iconically, as well as the variations among their gestures.

Related work

Grammar-based formalisms have become increasingly popular in research on vision-based activity recognition during the last 15 years (see Chanda and Dellaert 2004 for a survey). Decomposing complex behaviors into primitive ac-

tions (symbols), and given ways of performing a behavior or an activity are both described as strings of symbols. The rules of a grammar model determine which combination of primitive actions comprises a valid performance of a behavior. There are several reasons why a grammar model is appealing to the representation of complex activity patterns. Grammar can be elegantly represented, its structure is interpretable, and it can be used to formulate concise descriptions of action patterns. Many studies have applied syntactic approaches to recognizing different types of nonverbal behavior, i.e. the Finite State Machine (FSM) for hand gesture recognition (Hong, Turk, and Huang 2000) or Context-Free Grammar when used to represent and recognize human actions and interactions (Ryoo and Aggarwal 2006; Kitani, Sato, and Sugimoto 2006).

To address the uncertainty created by noisy sensors or computer vision, syntactic approaches have been extended to include probabilities early on. Stochastic Context-Free Grammar (SCFG) (Stolcke 1994) have been applied to different vision-based applications such as simple hand gesture recognition Ivanov and Bobick (2000) or surveillance in parking. Most of these approaches have used the Earley-Stolcke parsing algorithm (Stolcke 1994) for efficient probabilistic parsing. Minnen, Essa, and Starner (2003) use similar techniques for activity recognition during the Tower of Hanoi task, and Moore and Essa (2002) to recognize multi-agent activities in blackjack card games. All of these systems define the applied grammar syntax manually and have applied it only for task recognition at levels of rather complex actions with clear-cut compositionality with respect to both units and structures. None faced the challenge of learning a grammar from samples of action performances. There are two exceptions. They are work by Kitani, Sato, and Sugimoto (2006) on very simple activity recognition, and by Zhang, Tan, and Huang (2011) who learned SCFGs for applications such as the recognition of gymnastic exercises, traffic events and multi-agent interactions.

The idea of attributed grammars was originally propose by Knuth in 1968 for linguistic frameworks. The idea of combining statistical and syntactic approaches using linguistic frameworks for nonlinguistic applications can be traced back to the 80's, when Tsai and Fu (Tsai and Fu 1980; Fu 1986) proposed attributed grammars as a means to integrate statistical regularities into syntactic pattern analysis. However, many of the applied grammar formalisms in behavior analysis have not integrated statistical and syntactic aspects within the same approach. That said, they have processed data at two separate levels. The first is low-level segmentation and symbolization, where statistical methods such as HMMs are used (Zhang, Huang, and Tan 2006; Chen, Georganas, and Petriu 2008; Ivanov and Bobick 2000). The Second involves the high-level recognition of longer range pattern with the aid of syntactic methods such as SCFG. Exceptions to this method are studies that have proposed attributed grammar formalisms in context such as activity recognition (Damen and Hogg 2009) or the detection of abnormal events when parking a car (Joo and Chellappa 2006).

Definition of Feature-Based Stochastic Context-Free Grammar

Our proposed framework, FSCFG, is an extension of the probabilistic SCFG framework proposed by Stolcke (1994). An SCFG is defined by the following symbol sets:

- Σ, a finite set of terminal symbols.
- $\mathcal{N}$, a finite set of non-terminal symbols.
- $S \in \mathcal{N}$, a start symbol.
- $\mathcal{R}$, a finite set of rules, each of the form $X \rightarrow \lambda$ with the left-hand side $X \in \mathcal{N}$ and the right-hand side $\lambda \in (\mathcal{N} \cup \Sigma)^*$. Each rule is associated with a probability $P \in [0, 1]$ (shown in brackets in front of each rule).

Based on this definition, an FSCFG is defined by the following additions (see Figure 1):

- $\mathcal{F}$, a finite set of n features $\{f_1, \ldots, f_n\}$. A feature set is defined as $\mathbf{F}$=$\{f_1$=$v_1, \ldots, f_n$=$v_n\}$.
- Each terminal $t \in \Sigma$ is represented by a weighted set of l feature sets: t=$\{(\mathbf{F}_i, w_{\mathbf{F}_i}) \mid i$=$1, \ldots, l;\ w_{\mathbf{F}_i} \in]0, 1[\ \}$.
- Each feature of a terminal is weighted for all its feature sets equally, given by $\{(f_i, w_{f_i}) \mid i = 1, \ldots, n; w_{f_i} \in \mathbb{R}\}$.

Through the definition of n features used to form l feature sets, FSCFG allows for feature-based representation of data samples in an n-dimensional feature space. On this basis, as illustrated in Figure 1, each terminal is not represented as an atomic symbol, but as the prototype of a cluster with n features. The importance of the i-th sample (i.e. feature set) to the cluster is determined by $(\mathbf{F}_i, w_{\mathbf{F}_i})$, and the importance of the i-th feature within a cluster is determined by (f_i, w_{f_i}).

Accordingly, a given string of symbols to an FSCFG is an ordered set of n-dimensional samples. For instance, in the case of working with movement streams, a symbol corresponds to the feature-based representation of a movement segment. In this way, statistical (feature-based) and syntactic (structure-based) processing are unified within a hybrid framework that learns not only rule-based syntactic models of symbols, but also takes into account the statistical relations in the underlying features spaces.

Parsing with FSCFG

In contrast to SCFG, FSCFG can compute the similarity between two symbols (or terminals) in an n-dimensional feature space. Hence, while parsing with an FSCFG, the match between a terminal against an input symbol is computed probabilistically, depending on the measured similarity between the parsing terminal and the parsed input symbol in their n-dimensional feature space. Specifically, parsing the i-th symbol $\mathbf{x}_i = \{\mathbf{F}_{x_i}\}$ of an input string $\mathbf{x} = \{\mathbf{x}_1, \ldots, \mathbf{x}_m\}$, through a terminal t=$\{(\mathbf{F}_1, w_{\mathbf{F}_1}), \ldots, (\mathbf{F}_l, w_{\mathbf{F}_l})\}$ can be measured probabilistically (as opposed to binary true/false matching in SCFG). The parsing probability is given by

$$p(\mathbf{x}_i|t) = \sum_{j=1}^{l} w_{\mathbf{F}_j}\ g(\mathbf{x}_i|\mathbf{F}_j), \qquad (1)$$

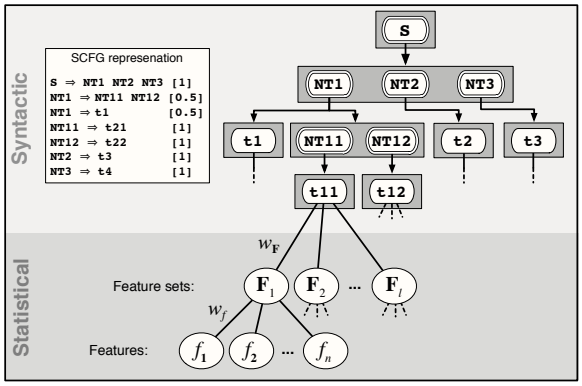


Figure 1: Hybrid model of FSCFG, where terminal symbols in the syntactic structure serve as the interface to the statistical feature-based representations.

where $g(\mathbf{x}_i|\mathbf{F}_j)$ is a multidimensional Gaussian distribution. In order to keep this computational step simple and efficient, we make the naïve assumption that the features are statistically independent of each other. Hence, the covariance matrix between n features is diagonal, and in this way the multidimensional Gaussian is defined as product of feature-specific Gaussians:

$$g(\mathbf{x}_i|\mathbf{F}_j)=\prod_{k=1}^{n} w_{f_k}\ gauss(f_{k,\mathbf{x}_i}|\mu_{f_k},\sigma_{f_k}), \qquad (2)$$

where the mean of Gaussian function μ_{f_k} is set to the feature value of $f_{k,\mathbf{F}_j}$; and the standard deviation σ_{f_k} can be set for each feature separately (i.e. equal to the standard deviation of each feature in the training data). As a result of applying this equation, the parsing probability computed in a "scanning step" indicates how well a symbol is parsed by a terminal. Thus in FSCFG, the scanning step of $\mathbf{x}_i$ through t multiples both forward and inner probabilities with $p(\mathbf{x}_i|t)$.

Besides parsing, an FSCFG grammar can be learned from samples of strings, in a supervised manner by exploring both spaces of possible structures and parameter values. Learning the structure of an FSCFG corresponds to the process of finding the optimal set of rules. The parameters that need to be optimized during learning are P (the probability of each rule), $w_{\mathbf{F}}$ (the weight of each feature set), and w_f (the weight of each feature) for each terminal.

Learning the structure of an FSCFG

In order to find an optimal set of rules, first an initial set of rules are generated which fit a given sample of strings maximally. To this end, for each symbol in the string, we generate a terminal with a single feature set $t{=}\{(\mathbf{F}_1, w_{\mathbf{F}_1})\}$. In addition, a non-terminal X and a lexical rule $X \rightarrow t\ [1]$ is generated for each terminal. In the end, a start rule $S \rightarrow X\ Y \ldots [1]$ comprising the entire given string – by producing the sequence of all non-terminals – is added to the set of rules.

Upon initialization, the structure is generalized by applying the *merge* and *chunk* operators proposed by Stolcke. The merge operator $merge(X_1, X_2) = Y$ replaces all occurrences of the non-terminals X_1 and X_2 with a new non-terminal Y. The chunk operator $chunk(X_1 \ldots X_k) = Y$ replaces all occurrences of the ordered sequence of the non-terminals $X_1 \ldots X_k$ with a single new non-terminal Y and adds a new chunking rule $Y \rightarrow X_1 \ldots X_k$ to the grammar. These operators simplify the grammar by decreasing its *Description Length (DL)* (Rissanen 1983). The *loss measure* during this process is the negative logarithm of the Bayesian posterior parsing probability. The likelihood term, which indicates how well the given samples fit the learned model, is set to the parsing probability of the samples; the prior probability of a grammar is set to its DL, which is proportional to the code length in bits needed to store the grammar's rules. In this way, by using a Bayesian loss measure, the grammar structure is modified towards a trade-off between generalization (or simplicity) and fitting of the model to the given data.

Given the Bayeisan loss measure for an FSCFG, similarly to SCFG, different search strategies can be applied to find a relatively optimal grammar structure. (1) In *best-first search* all candidates of merge and chunk operators are pooled and the best one locally is selected. To overcome local minima, if the loss begins to increase, further modification steps are checked for a sudden drop of loss (look-ahead steps). (2) *Multilevel best-first search* is very similar to the best-first search, except that it searches for the best merge and chunk candidates at two different levels. Only after the best local merge candidate is selected, is the best local chunk candidate chosen. (3) In *beam search*, instead of always choosing the best local modification candidate, different possible sequences of merge and chunk operators are checked as a tree of possible moving paths in the structure space.

When providing samples to learn the structure of an FSCFG, the process of adding new rules to the grammar and optimizing them by searching the structure space, is repeated for each sample that cannot be parsed by the grammar with a high enough probability.

Learning the parameters of an FSCFG

The parameters of an FSCFG are learned and optimized during both learning the structure and parsing new input strings, after a default initialization at the beginning.

Rule probabilities (P): Similarly to SCFG, the probability of each rule is determined from how often the rule has been invoked for parsing, normalized by the sum of all invocations of the rules with the same left-hand side non-terminal.

Weights of features (w_f): The weights of features are set for each terminal individually, and they are set for all feature sets of a terminal equally. This means, w_f refers to the weight of feature f in the terminal t for all its l feature sets $\{\mathbf{F}_i|i = 1, \ldots, l\}$. w_f is defined inversely proportional to the standard deviation of the values of f among all feature sets, given by

$$w_f(t){=}\frac{1}{std(\{f \in \mathbf{F}_i|\mathbf{F}_i \in t; i = 1, \ldots, l\}) + 1} \qquad (3)$$

Hence, the higher the variance of a feature within a terminal, the less discriminative is the feature for the terminal and the less it contributes to the parsing through that terminal. In other words, during parsing, each terminal is more sensitive to its less variable features and in this way an FSCFG distinguishes between *variant and invariant features* for each terminal. The sensitivity of each terminal to the given features depends on the parsed input symbols and can lead to different weightings at different positions of the grammar rules. Thus, some variant features for a terminal may be counted as invariant for other terminals and vise versa.

Weights of feature sets ($w_{\mathbf{F}}$): Computing the weight of each feature set of a terminal employs a counter which is normalized by the sum of its values in each terminal. Initially, this counter is set to one yielding $t{=}\{(\mathbf{F}_1, \frac{1}{l}), \ldots, (\mathbf{F}_l, \frac{1}{l})\}$. This set of feature sets can be extended in two ways: (1) During parsing, when terminal t parses a symbol $\mathbf{x}_i$, the symbol – which is represented as a single feature set – is added to the terminal, with an initial counter of one. In this way, parsing reshapes the terminals of a grammar towards the features of the parsed symbols. (2) During learning the structure of a grammar, when merging two lexical non-terminals $merge(X_1, X_2) = Y$ with $X_1 \rightarrow\ t_1$ and $X_2 \rightarrow\ t_2$, the right-hand side terminals – e.g. $t_1{=}\{(\mathbf{F}_1, 1)\}$ and $t_2{=}\{(\mathbf{F}_2, 1)\}$ – are also merged. This results in a new rule $Y \rightarrow\ \ t$, where the new terminal t is a cluster of the old terminals: $t{=}\{(\mathbf{F}_1, \frac{1}{2}), (\mathbf{F}_2, \frac{1}{2})\}$. These two incorporation steps for feature sets during both parsing and learning may lead to terminals that are too large with too many feature sets and therefore additional computational costs. To increase the efficiency, we perform a pruning step that combines similar feature sets. In this case, the counter of a new feature set is set to the sum of the counters of the ones replaced. As a result, the new feature set gains more weight and therefore more influence in the representation of its terminal.

Through the computation and optimization of the parameters $w_{\mathbf{F}}$ and w_f during both learning and parsing, an FSCFG learns the set of its terminals continuously, dynamically and incrementally. This is a useful feature when dealing with continuous input strings without a clear definition of compositional elements as symbols, such as continuous human motion data. In many studies that have applied grammar-based approaches to learning models of human actions, a symbolization step has been performed as preprocessing for the input data. For instance, Wang, Lorette, and Bouthemy (1991) and Ivanov and Bobick (2000) applied Hidden Markov Models (HMMs) to learn symbols as prototypes of movement segments before applying a SCFG. In FSCFG, by representing each terminal as weighted feature sets, there is no clear cut off between these two processing levels and the statistical symbolization of terminals is homogeneously integrated in the learning process of syntactic structure. Furthermore, such an integrated symbolization process also takes the learned syntax of the grammar into account, and may lead to different symbolization results for different parts of a the grammar.

Handling uncertain input

An important challenge in grammar learning and parsing is uncertainty in the input data. This may lead to *deletion errors* when an expected symbol is missing in the input stream, *insertion errors* when symbols occur spuriously, or *substitution errors* when a symbol is parsed with the wrong terminal. Since FSCFG can parse any symbol by any terminal through feature-based parsing, the substitution error is handled implicitly. To deal with the insertion and deletion errors during parsing, we introduce special symbols (cf. Ivanov and Bobick 2000):

- A new terminal $skip \in \Sigma$ with no feature set
- A new non-terminal $SKIP \in \mathcal{N}$, mapped to the $skip$ terminal through the lexical rule $SKIP \rightarrow skip$.
- A new terminal $\epsilon \in \Sigma$ which is used to create null production rules of the form $X \rightarrow \epsilon$ for any $X \in \mathcal{N}$.

$skip$ handles insertion errors as it treats all symbols equally and is assigned a small probability. Consequently, a parsing path through $skip$ is costly and is only used if otherwise parsing would fail. To enable this path, we add to any lexical rule $X \rightarrow t$ the new alternative rule $X \rightarrow SKIP\ X$. In addition, the start rule of the form $S \rightarrow \lambda$ receives the alternative rule $S \rightarrow \lambda\ SKIP$. Finally, a new lexical rule $SKIP \rightarrow skip$ is added to the grammar. ϵ handles deletion errors because it forces its associated non-terminal to be ignored during parsing. Additionally, we add to each lexical rule $X \rightarrow t$, the alternative rule $X \rightarrow \epsilon$ with a small probability. In this way, the parsing algorithm ignores the occurrence of the terminal t only if need be.

Note that, from parsing to structure learning, deletion and insertion errors exchange their roles as cause and effect. Learning a grammar rule based on an input string with an insertion error results in adding an erroneous lexical rule which then causes a deletion error when parsing a correct string. On the other hand, when learning a rule from a string with a deletion error, the missing lexical rule in the learned grammar causes an insertion error when parsing a correct string. Using these error handling symbols when confronted with noisy data, the FSCFG framework should entertain both hypotheses that either the learned grammar structure is incorrect or the given input string is noisy. Hence, when a given input string uses a deletion handling rule such as $X \rightarrow \epsilon$, the structure of the grammar is optimized by adding an alternative for each rule containing X in the form of $Y \rightarrow \lambda X \nu$, by omitting the missed non-terminal as $Y \rightarrow \lambda\nu$. Furthermore, in the case of using a $skip$-rule, the corresponding terminal $skip$ and non-terminal $SKIP$ are renamed to new symbol names and are inserted into the set of grammar rules.

These error handling rules result in a growing FSCFG which also contains rules and symbols for noisy data. However, after providing enough training data, the noisy rules will be used less for parsing and will result in low probable rules. In a successive pruning step after learning, these rules can be removed from the grammar as needed.

3D Iconic Gestures Dataset

To test our grammar framework, we have recorded a dataset of 1739 iconic gestures performed by 29 participants (20

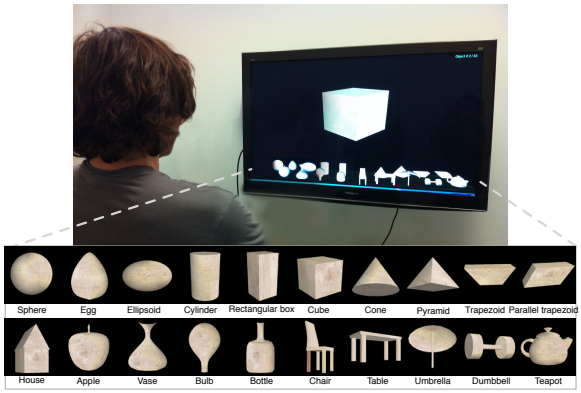


Figure 2: 3D object models used in the study; in this situation the cube is to be described next.

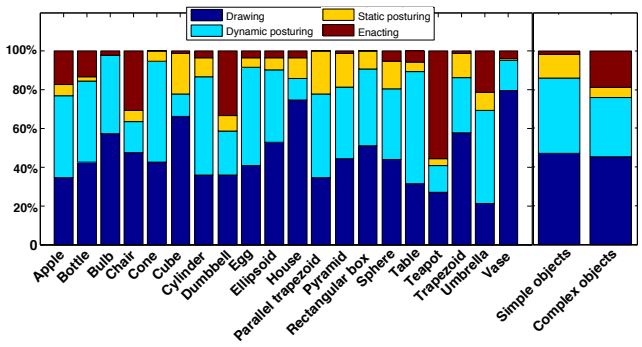


Figure 3: The rates of different representational techniques used to refer to each 3D virtual object.

males and 9 females; from 9 different declared ethnicities) to depict different objects. All gestures were recorded using the MS Kinect™. Participants were presented with twenty virtual 3D models of simple and complex objects on a screen (see Figure 2). One of the objects moved to the center of the screen, enlarged and rotating, and participants signaled when they felt ready to perform a gesture for that object. Then, the object disappeared and a photo of an addressee person was shown to whom the object had to be depicted gesturally. After the participants retracted their hands or gave a verbal signal, the face disappeared and the next object was shown (order randomized). The participants were told that their gestures would be videotaped and shown to other participants, who would have to recognize the object from the same set. Each object was shown three times. This procedure resulted in approximately 87 gestures per object, each recorded in color video, depth video and 3D motion of the skeleton (in 30 fps). This *three-dimensional iconic gesture dataset (3DIG)* has been made available online[1].

Analysis of the gestures performed

Analyses of the video data revealed that the participants used four different representational techniques to depict visual or functional features for each object: (1) *Drawing* the 3D or 2D contours of an object in the air. (2) *Enacting* an action performed on an imaginary object (e.g. "throwing" to refer to the ball). (3) *Static posturing* with the hand(s) held still in order to form an object (e.g. forming a concave circular shape with both hands). (4) *Dynamic posturing* where drawing and static posturing are combined, as if the gesturer is touching the surface of an imaginary object. Figure 3 shows the use of each technique among gestures performed for each object or object set. Note that within a single gesture performance, different techniques could be used sequentially or simultaneously. As shown, the drawing technique was the most dominant choice followed by dynamic posturing. Static posturing was used more frequently for simple objects with abstract geometrical shapes, whereas the enacting technique was used preferentially for complex everyday objects.

Structural variability	**Examples of variation**
Degree of simplification	Drawing a 3D shape or a 2D projection.
Ordering	First, referring to the triangle shape of a cone and then to its circular bottom, or vise versa.
Repetition	Drawing the round shape of a circle once, twice or three times.
Handedness	Drawing a circle with one hand or both hands.
Feature-based variability	**Examples of variation**
Direction	Drawing while moving a hand upward or downward, clockwise or counter-clockwise.
Velocity	Drawing fast or slow.
Size	Drawing a small or a big circle.
Position	Drawing in front of head or chest.
Projection	Drawing a horizontal or vertical projection.
Form	Making curved movements or straight ones.

Table 1: Structural and feature-based variabilities among iconic gesture performances.

Except for static posturing, the wrists movements made the greatest contribution to the gestural depiction. We thus concentrate in the following section on wrist trajectories in gesture space, which can be better captured with low-cost tracking systems such as MS Kinect ™ than, e.g., hand-shape. Our goal is to learn a hybrid grammar for iconic gesture trajectories.

The structural property most commonly acknowledged for gestures is the division of a gesture into different phases (Kendon 1972): (1) *pre-stroke preparation* to move the hands from a rest position to a start position; (2) *stroke*, the meaningful and most effortful phase; (3) *post-stroke retraction* to move the hand back to a rest position. Furthermore, even during the stroke phase, some parts of the movement might be transitional (e.g. when continuing to draw in a different location). This means that in order to interpret a communicative iconic gesture, the irrelevant parts (i.e. pre-

[1] http://projects.ict.usc.edu/3dig/

stroke, post-stroke, transitional sub-movements) need to be ignored. This structural property should be identified when learning a grammar for gestures.

To get a better idea of the variabilities involved in gesture, we began by analyzing the variation among the gestures observed in the corpus. Table 1 reports the most prominent structural (or syntactic) variabilities, which can lead to very different gestures performed for the same object. Below the level of syntactic variation, there are spatiotemporal variations of features that represent statistical variabilities. At this level, the spatiotemporal features can be *invariant* and thus characteristic for a specific gesture class (e.g., a curved trajectory for round objects), or they can be variant and thus likely irrelevant with respect to a specific technique (e.g. movement direction while drawing). In the next section, we report on how FSCFG is able to cope with both kinds of variability while generalizing over different gestures, separating different performing ways and determining variant and invariant features from seemingly meaningless aspects of a gesture.

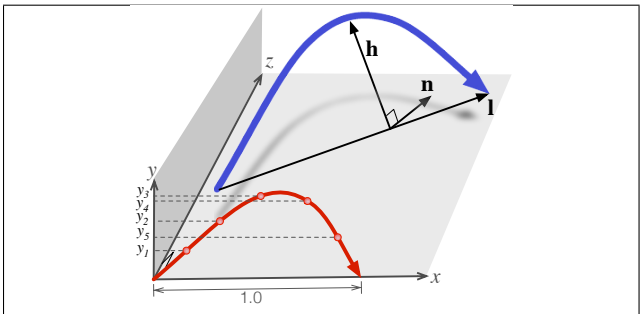
Results

We applied the FSCFG framework to learn a generalized model from the gesture performances performed for each of the twenty objects in the 3DIG dataset. The learned grammar models were then tested by classifying the unseen gesture performances, based on their parsing probabilities.

Before learning a grammar model of hand gestures, the continuous wrist movement trajectories needed to be segmented into discrete symbols. Inspired by the concept of guiding strokes (Kopp and Wachsmuth 2004) as segments of movement trajectories of hand gestures, we segmented the continuous trajectories on the minima of their velocity profiles. Since wrist movements are slower while changing direction or drawing a sharp curve, the resulting segments were in the form of simple curves or straight lines. Then, each segment was normalized as follows (see the figure in Table 2). First, the start position of the segments was translated to the origin. Second, the segments were rotated about all three spatial axes and mapped on the x-y plain. Finally, segments were resized while maintaining their width-height proportions. The normalized trajectory was then resampled at equal distances on the x axis, whose y coordinates represented the form of the trajectory in 5 dimensions. As shown in Table 2, each segment is represented by 18 features (as a single feature set). These extracted features reflect the statistical variabilities that were shown in Table 1.

Learning FSCFG models of gestures

One FSCFG model was learned for all gesture performances for each object. At the beginning of the learning process for each model, a small subset of gestures (e.g. three performances) was used to initialize a maximally fitting FSCFG. For this purpose, the following set of symbols and rules were added to the grammar: a terminal symbol for each segment, a lexical rule to produce each terminal, a rule producing the movement sequence for each hand, and a start rule producing the whole gesture as a sequence of left and right hand non-terminals successively.



Features	Dim. nr.	Calculation/Notion
Samples' heights	5	$(y_1, y_2, y_3, y_4, y_5)$
Start to end vector	3	$\mathbf{l} = (x_l, y_l, z_l)$
Bounding box	2	$(\| \mathbf{l} \|, \| \mathbf{h} \|)$
Normal vector	3	$\mathbf{n} = (x_n, y_n, z_n)$
Direction of concavity	3	$\in \{-1, 1\}$ for each dim.
Average speed	1	$\| \mathbf{l} \| \ / \ duration$
Start time	1	start time of movement

Table 2: Extracted features from each movement segment. The blue arrow at the top represents a segment of a wrist movement trajectory, and the red arrow shows the same trajectory after normalization.

After this initial batch mode, the remaining samples were given to the algorithm one by one in online mode. In this mode, each training sample is first parsed, and if the parsing probability is above a given threshold, the grammar rules are adopted in three steps: (1) updating the rule probabilities, (2) adding the parsed segments as new feature sets to the parsing terminals, and (3) updating the weights of features and feature sets ($w_{\mathbf{F}}$ and w_f). In case the parsing probability is below the threshold, first the grammar is extended by the new rules to fit the given sample maximally. Consequently, the resulting suboptimal structure is optimized by applying merge and chunk operators, and in this way, searching the structure space according to the Bayesian loss measure. In this application of FSCFG, applying the naïve but fast best-first search algorithm with two look-ahead steps, we could achieve classification results as accurate as the multilevel best-first and beam search algorithms.

Figure 4 shows an example of an FSCFG model learned from drawing gestures performed for the sphere object. The grammar possesses three start rules that represent three main ways of performing a gesture for sphere. Each of the start rules produces one or two non-terminals, whereas each of the non-terminals produces the movement of either the left or right wrists as strings of lexical non-terminals. These strings represent more specific ways of performing the same gesture.

Each lexical non-terminal produces a terminal that considers the regularities at the level of features. As illustrated, each terminal is the prototype of a group of feature sets, where the strength of each connection (i.e. $w_{\mathbf{F}}$) indicates the weight of the corresponding feature set for each terminal. It can be seen that some of the terminals link to one

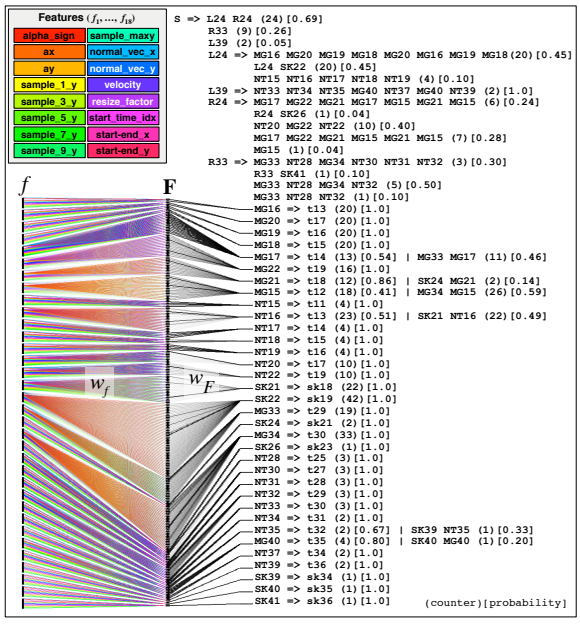


Figure 4: An example of a learned FSCFG for sphere.

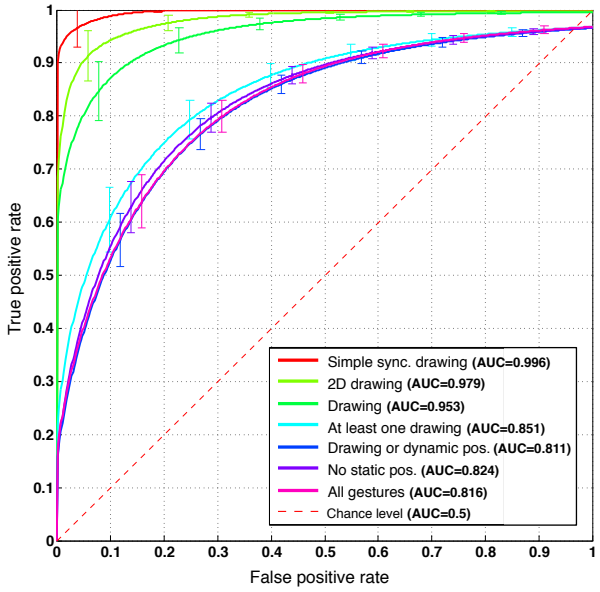


Figure 5: The ROC curves of classification using FSCFGs. A ROC curve represents the mean of twenty ROCs, each representing the classification result of gestures for an object.

or two strong feature sets (such as MG15, MG21 and MG22) that they represent their typical movement segments. In contrast, non-terminals such as MG17 or MG40 are represented through a set of equally influential feature sets. These terminals represent parts of a gesture that vary widely and are thus most likely less informative (e.g. the pre-stroke or post-stroke phases). At the next level of statistical regularities, feature sets are connected to the same features but with different weights (i.e. w_f). As illustrated, some of the features have a higher impact on their feature sets and consequently on the representation of their terminals than others. Small sets of highly-weighted features thereby represent "invariant" features, i.e. features that are most characteristic of a particular terminal (and thus a specific part of a gesture).

In sum, the feature-based representation of terminals generalizes over many spatiotemporal deviations in different parts of gesture performance. Grounded in this statistical representation, the syntactic grammar rules generalize over the structural variants among different gesture performances for the same object.

Classification results

To evaluate the FSCFG models of gestures quantitatively, we used them to probabilistically parse unseen gestures and to classify them according to the highest parsing probability. To this end, we divided the given corpus into different subsets of gesture performances based on a manual annotation of the gesture videos. Since the representation of gestures as spatial wrists movement trajectories is an underspecified representation of iconic gestures, it was expected that the classification algorithm would achieve relatively more accurate results for gestures performed with drawing technique than for example with static postures, where the absent hand postures play a decisive role. We tested the classification results for the following subsets of gestures: (1) The subset of *simple synchronized drawing gestures* (367 gestures) in which the drawing technique was used, both hands moved synchronously, no part of the gesture was repeated and a two-dimensional projection of the referred object was depicted; (2) *2D drawing* subset (543 gestures) with only drawing gestures with two-dimensionally depicted shapes; (3) *Drawing* subset (702 gestures) that contained gestures performed only with the drawing technique; (4) The subset *at least one drawing* (909 gestures) refers to all gestures with at least one part performed with drawing technique; (5) *Drawing or dynamic posturing* subset (1303 gestures) consisted of all gestures performed only with one of these techniques; (6) *No static posturing* (1507 gestures) was the subset of gestures in which no static posturing technique was applied; and finally (6) *all gestures* contained all 1739 gestures in the 3DIG corpus.

Figure 5 shows the receiver operating characteristics (ROC) graphs of two-fold cross-validation results, for each subset of the gesture performances. The results show that FSCFG classified drawing with relatively high performance. A considerable drop in classification accuracy occured when the gesture performances with underspecified techniques were added to the dataset.

Figure 6 shows the confusion matrix of the classification results of the drawing subset. Many of the confusions occur between gestures for objects that share many visuospatial properties. For instance, many of the gestures for sphere and ellipsoid, cube and rectangular box, or cone and pyramid are in fact performed ambiguously, because of a too rough depiction or their similar 2D projection. These con-

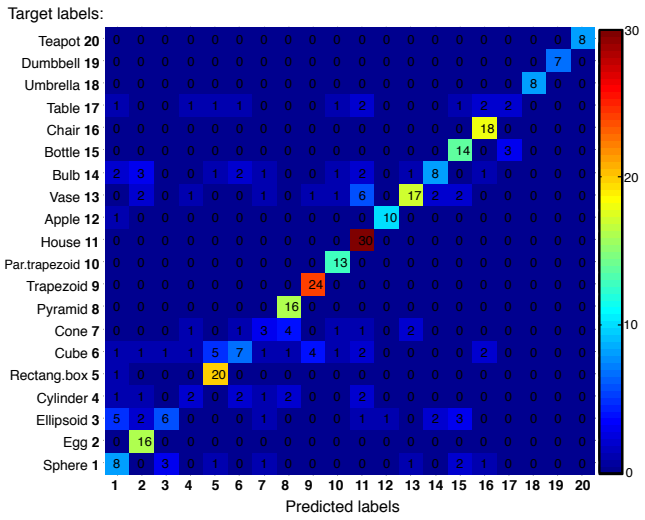


Figure 6: Confusion matrix of the FSCFG classification results, performed on drawing gestures.

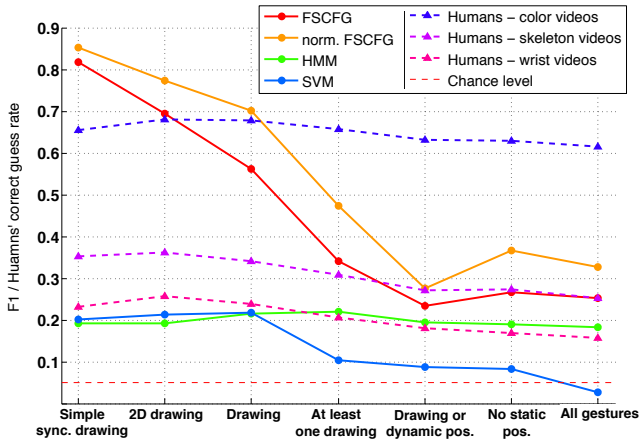


Figure 7: Comparing the classification performance (F1) of FSCFG to different methods and human judgment baselines, given different subsets of gestures.

fused classes are in fact a result we had hoped for, since the eventual goal is to learn a grammar that carves out gestural strategies to iconically depict object properties and not specific individual objects. For example, a single FSCFG model for all performances for sphere and ellipsoid can be learned to represent a generative gesture model for round objects. Other confusions can be found with complex objects such as the table object, which are referred to by relatively very few but diverse drawing gestures, resulting in very different training and test samples.

In order to evaluate the classification performance of the FSCFG models, we compared them to the performance of other methods on the same datasets (Hidden Markov Models[2] and Support Vector Machines[3]), and human judgment performance (see Figure 7). The features we used for HMM were instantaneous features of movement trajectories, such as spatial position, velocity and acceleration at each time step for each wrist. As features for SVM, we took the histogram of the features of all segments of a given gesture sample. The idea of these so-called HoGS features was proposed in a previous work on this dataset (Sadeghipour, Morency, and Kopp 2012), in which – in contrast to this application – the best features were selected for each pair of SVMs separately. As shown in Figure 7, the FSCFG model outperforms the other algorithms in all subsets of the gesture dataset. Moreover, we found that normalizing the parsing probability of each class for each gesture performance to the sum of all parsing probabilities from all twenty FSCFG models improved the classification accuracy significantly, as shown in Figure 7 under the notion of "norm. FSCFG".

Human judgment for recognizing the performed gestures represents a base-line for this classification task. We carried out an online study in which the participants were asked to guess which of the twenty objects each gesture refered to. Each of the 240 participants watched 10 color videos of gesture performances, then 10 videos showing rendered skeletons of the gesturers, and finally 10 videos showing only the wrists movements as two moving dots. We excluded participant who did not complete the task to the end, or who answered at least one test question incorrectly[4]. The remaining 179 participants together guessed each gesture performance in each of the three demonstration conditions at least once. Notably, as shown in Figure 7, normalized FSCFG models achieved better recognition rates than humans in the first three subsets of drawing gestures, for which the extracted features were designed. Further, the performance of normalized FSCFGs is in all subsets was better than human judgments in the skeleton or wrist conditions.

Conclusion

We have presented a hybrid grammar-based approach to capture the structural and feature-based characteristics of natural iconic gestures. To address the large variability and weak compositionality in gesturing, our FSCFG models simultaneously leverage low-level statistical regularities and high-level syntactic patterns during learning and recognition. Furthermore, the extension of the parsing algorithm to deal with uncertain input allowed for learning of noisy human motion data. We extracted features from wrist movement trajectories and achieved reliable classification results of drawing gestures, in which wrist movements depicted the contours of objects. The FSCFG models of these gestures are generalized interpretable representations of their samples. In classification, these models not only outperformed other classi-

[2]Using the Bayes Net Toolbox for Matlab, available at http://code.google.com/p/bnt/, with 5 hidden states and 8 Gaussian mixtures.

[3]Using the LIBSVM library (Chang and Lin 2011), available at http://www.csie.ntu.edu.tw/ cjlin/libsvm/, with ν-SVM type (ν=0.01) and radial basis kernel type (γ=0.01).

[4]Three videos were shown randomly, in which the participants were asked to click on a specific object in textual form.

fication methods in gesture recognition, but they were also more accurate than human judgment on color videos of the drawing gestures.

Acknowledgements This research is supported by the Deutsche Forschungsgemeinschaft (DFG) in the Center of Excellence EXC 277 in 'Cognitive Interaction Technology' (CITEC).

References

Bobick, A., and Ivanov, Y. 1998. Action recognition using probabilistic parsing. In *Computer Vision and Pattern Recognition, 1998 IEEE Computer Society Conference on*, 196–202.

Chanda, G., and Dellaert, F. 2004. Grammatical methods in computer vision: An overview. Technical report, Georgia Institute of Technology.

Chang, C.-C., and Lin, C.-J. 2011. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology* 2:27:1–27:27.

Chen, Q.; Georganas, N. D.; and Petriu, E. 2008. Hand gesture recognition using haar-like features and a stochastic context-free grammar. *Instrumentation and Measurement, IEEE Transactions on* 57(8):1562–1571.

Damen, D., and Hogg, D. 2009. Attribute multiset grammars for global explanations of activities. In *BMVC*, 1–11.

Fu, K. S. 1986. A step towards unification of syntactic and statistical pattern recognition. *IEEE Trans Pattern Anal Mach Intell* 8(3):398–404.

Großekathöfer, U.; Sadeghipour, A.; Lingner, T.; Meinicke, P.; Hermann, T.; and Kopp, S. 2012. Low latency recognition and reproduction of natural gesture trajectories. In *ICPRAM (2)'12*, 154–161.

Hong, P.; Turk, M.; and Huang, T. S. 2000. Gesture modeling and recognition using finite state machines. In *Automatic Face and Gesture Recognition, 2000. Proceedings. Fourth IEEE International Conference on*, 410–415. IEEE.

Ivanov, Y., and Bobick, A. 2000. Recognition of visual activities and interactions by stochastic parsing. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 22(8):852–872.

Joo, S.-W., and Chellappa, R. 2006. Attribute grammar-based event recognition and anomaly detection. In *Computer Vision and Pattern Recognition Workshop, 2006. CVPRW '06. Conference on*, 107–107.

Kendon, A. 1972. *Some relationships between body motion and speech*. New York: Pergamon Press.

Kitani, K. M.; Sato, Y.; and Sugimoto, A. 2006. An mdl approach to learning activity grammars. Technical Report 376, IEICE - The Institute of Electronics, Information and Communication Engineers, Tokyo, Japan.

Knuth, D. E. 1968. Semantics of context-free languages. *Mathematical systems theory* 2(2):127–145.

Kopp, S., and Wachsmuth, I. 2004. Synthesizing multimodal utterances for conversational agents: Research articles. *Comput. Animat. Virtual Worlds* 15(1):39–52.

Minnen, D.; Essa, I.; and Starner, T. 2003. Expectation grammars: leveraging high-level expectations for activity recognition. In *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, volume 2, 626–632.

Moore, D., and Essa, I. 2002. Recognizing multitasked activities from video using stochastic context-free grammar. In *Eighteenth national conference on Artificial intelligence*, 770–776. Menlo Park, CA, USA: American Association for Artificial Intelligence.

Pastra, K., and Aloimonos, Y. 2012. The minimalist grammar of action. *Philosophical transactions of the Royal Society of London. Series B, Biological sciences* 367(1585):103–117.

Ramamoorthy, A.; Vaswani, N.; Chaudhury, S.; and Banerjee, S. 2003. Recognition of dynamic hand gestures. *Pattern Recognition* 36(9):2069–2081.

Rissanen, J. 1983. A universal prior for integers and estimation by minimum description length. *Annals of Statistics* 11(2):416–431.

Ryoo, M. S., and Aggarwal, J. 2006. Recognition of composite human activities through context-free grammar based representation. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 2, 1709–1718.

Sadeghipour, A.; Morency, L.-P.; and Kopp, S. 2012. Gesture-based object recognition using histograms of guiding strokes. Proceedings of the British Machine Vision Conference, 44.1–44.11. BMVA Press.

Stolcke, A. 1994. *Bayesian Learning of Probabilistic Language Models*. Ph.D. Dissertation, University of California at Berkeley, Berkeley, CA.

Tsai, W. H., and Fu, K. S. 1980. Attributed grammar - a tool for combining syntactic and statistical approaches to pattern-recognition. *Ieee Transactions on Systems Man and Cybernetics* 10(12):873–885.

Turaga, P.; Chellappa, R.; Subrahmanian, V. S.; and Udrea, O. 2008. Machine recognition of human activities: A survey. *Ieee Transactions on Circuits and Systems for Video Technology* 18(11):1473–1488.

Wang, J.; Lorette, G.; and Bouthemy, P. 1991. Analysis of human motion: A model-based approach. In *7th Scandinavian Conference on Image Analysis, Aalborg*.

Yamato, J.; Ohya, J.; and Ishii, K. 1992. Recognizing human action in time-sequential images using hidden markov model. In *Computer Vision and Pattern Recognition, 1992. Proceedings CVPR'92., 1992 IEEE Computer Society Conference on*, 379–385. IEEE.

Yoon, H. S.; Soh, J.; Bae, Y. J.; and Yang, H. S. 2001. Hand gesture recognition using combined features of location, angle and velocity. *Pattern Recognition* 34(7):1491–1501.

Zhang, Z.; Huang, K.; and Tan, T. 2006. Complex activity representation and recognition by extended stochastic grammar. In *Proceedings of the 7th Asian conference on Computer Vision - Volume Part I*, ACCV'06, 150–159. Berlin, Heidelberg: Springer-Verlag.

Zhang, Z.; Tan, T.; and Huang, K. 2011. An extended grammar system for learning and recognizing complex visual events. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 33(2):240–255.