



vis VAST • INFOVIS • SCIVIS  
BIOVIS • LDAV  
2013

# A Systematic Review on the Practice of Evaluating Visualization

Tobias Isenberg, Petra Isenberg, Jian Chen,  
Michael Sedlmair, and Torsten Möller

*inria* informatics mathematics

UMBC

AN HONORS UNIVERSITY IN MARYLAND



universität  
wien

# Motivation

- research question: state of evaluation work in visualization?
- most common evaluation goals/methods?
- evaluation of what part of visualization process?
- evaluation done similarly in different sub-areas of visualization?
- history and current trends?

# Contributions

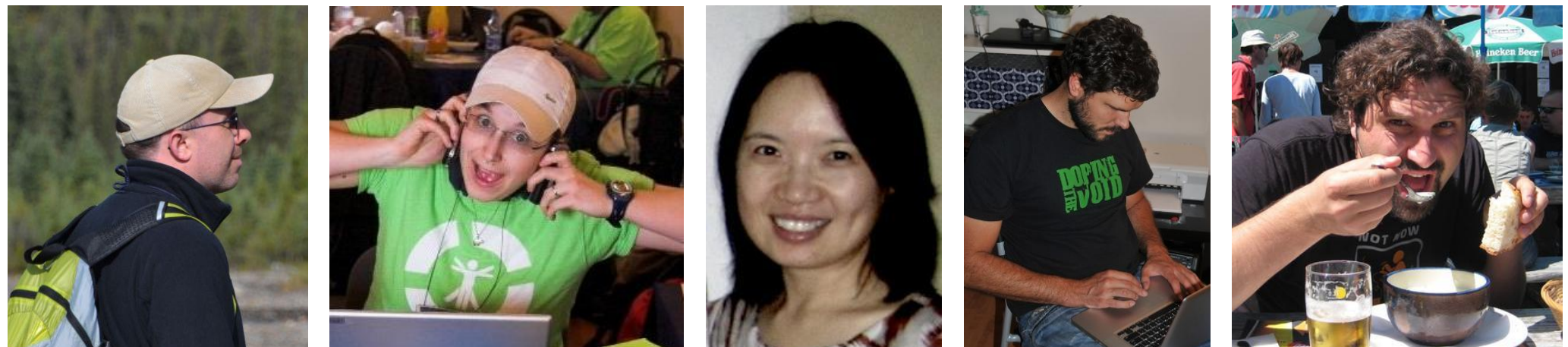
1. classification of evaluation use in “scientific visualization”
2. historical perspective of evaluation in visualization
3. considerations for improvement of evaluation in visualization

# Our approach

- literature review of IEEE Visualization/Scientific Visualization
- 581 papers



- coding by the 5 co-authors



# Related work

- Lam et al. [2012]: state of evaluation in “information visualization”
  - 850 papers of 1995–2010
  - InfoVis, VAST, EuroVis, Information Visualization Journal
  - 7 scenarios of evaluation goals

## Empirical Studies in Information Visualization: Seven Scenarios

Heidi Lam, Enrico Bertini, Petra Isenberg, Catherine Plaisant, and Sheelagh Carpendale

**Abstract**—We take a new, scenario-based look at evaluation in information visualization. Our seven scenarios, evaluating visual data analysis and reasoning, evaluating user performance, evaluating user experience, evaluating environments and work practices, evaluating communication through visualization, evaluating visualization algorithms, and evaluating collaborative data analysis were derived through an extensive literature review of over 800 visualization publications. These scenarios distinguish different study goals and types of research questions and are illustrated through example studies. Through this broad survey and the distillation of these scenarios, we make two contributions. One, we encapsulate the current practices in the information visualization research community and, two, we provide a different approach to reaching decisions about what might be the most effective evaluation of a given information visualization. Scenarios can be used to choose appropriate research questions and goals and the provided examples can be consulted for guidance on how to design one’s own study.

**Index Terms**—Information visualization, evaluation.

### 1 INTRODUCTION

EVALUATION in information visualization is complex since, for a thorough understanding of a tool, it not only involves assessing the visualizations themselves, but also the complex processes that a tool is meant to support. Examples of such processes are exploratory data analysis and reasoning, communication through visualization, or collaborative data analysis. Researchers and practitioners in the field have long identified many of the challenges faced when planning, conducting, and executing an evaluation of a visualization tool or system [10], [41], [54], [63]. It can be daunting for evaluators to identify the right evaluation questions to ask, to choose the right variables to evaluate, to pick the right tasks, users, or data sets to test, and to pick appropriate evaluation methods. Literature guidelines exist that can help with these problems but they are almost exclusively focused on methods—“structured as an enumeration of methods with focus on *how* to carry them out, without prescriptive advice for *when* to choose between them.” ([54, p.1], author’s own emphasis).

This paper takes a different approach: instead of focusing on evaluation methods, we provide an in-depth

discussion of evaluation scenarios, categorized into those for understanding data analysis processes and those which evaluate visualizations themselves.

The scenarios for understanding data analysis are

- Understanding environments and work practices (UWP),
- evaluating visual data analysis and reasoning (VDAR),
- evaluating communication through visualization (CTV), and
- evaluating collaborative data analysis (CDA).

The scenarios for understanding visualizations are

- Evaluating user performance (UP),
- evaluating user experience (UE), and
- evaluating visualization algorithms (VA).

Our goal is to provide an overview of different types of evaluation scenarios and to help practitioners in setting the right evaluation goals, picking the right questions to ask, and to consider a variety of methodological alternatives to evaluation for the chosen goals and questions. Our scenarios were derived from a systematic analysis of 850 papers (361 with evaluation) from the information visualization research literature (Section 5). For each evaluation scenario, we list the most common evaluation goals and outputs, evaluation questions, and common approaches in Section 6. We illustrate each scenario with representative published evaluation examples from the information visualization community. In cases where there are gaps in our community’s evaluation approaches, we suggest examples from other fields. We strive to provide a wide coverage of the methodology space in our scenarios to offer a diverse set of evaluation options. Yet, the “Methods and Examples” lists in this paper are not meant to be comprehensive as our focus is on choosing among evaluation scenarios. Instead, we direct the interested reader

- H. Lam is with Google, Inc, Mountain View, CA. E-mail: heidi.lam@gmail.com.
- E. Bertini is with the Department of Computer and Information Science, University of Konstanz, Box 78, Konstanz 78457, Germany. E-mail: enrico.bertini@uni-konstanz.de.
- P. Isenberg is with INRIA, Université Paris-Sud, Team Aviz, Bat 650, Saclay, Orsay Cedex 91405, France. E-mail: petra.isenberg@inria.fr.
- C. Plaisant is with the University of Maryland, 2117C Hornbake South Wing, College Park, MD 20742. E-mail: plaisant@cs.umd.edu.
- S. Carpendale is with the Department of Computer Science, University of Calgary, 2500 University Dr. NW, Calgary, AB T2N 1N4, Canada. E-mail: sheelagh@ucalgary.ca.

Manuscript received 8 Sept. 2010; revised 6 Nov. 2011; accepted 9 Nov. 2011; published online 30 Nov. 2011.

Recommended for acceptance by C. North.

For information on obtaining reprints of this article, please send e-mail to: tvcg@computer.org, and reference IEEECS Log Number TVCG-2010-09-0224. Digital Object Identifier no. 10.1109/TVCG.2011.279.



# Coding scheme (based on Lam et al. [2012])

- evaluating communication through visualization
- evaluating collaborative data analysis

# Coding scheme (based on Lam et al. [2012])

- evaluating communication through visualization
- evaluating collaborative data analysis
- understanding work practices
- visual data analysis and reasoning

# Coding scheme (based on Lam et al. [2012])

- evaluating communication through visualization
- evaluating collaborative data analysis
- understanding work practices
- visual data analysis and reasoning
- user performance
- user experience



# Coding scheme (based on Lam et al. [2012])

- evaluating communication through visualization
- evaluating collaborative data analysis
- understanding work practices
- visual data analysis and reasoning
- user performance
- user experience
- algorithmic performance (was **visualization algorithms**)

# Coding scheme (based on Lam et al. [2012])

- evaluating communication through visualization
- evaluating collaborative data analysis
- understanding work practices
- visual data analysis and reasoning
- user performance
- user experience
- algorithmic performance (was **visualization algorithms**)
- **qualitative result inspection**



# Results: evaluation scenarios (selection)

algorithmic performance  
35% of scenarios

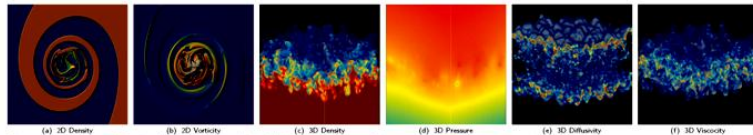


Fig. 1. Visualizations of 2D data (as pseudocolored height fields) and 3D data (volume rendered) used in our experiments.

name	data set					compressed size (MB) and compression time (seconds)											
	unique (%)	entropy (bits)	range (bits)	min	max	size (MB)	time (sec)	zlib	[RKB2006]	[EFF2000]	[ILS2005]	new scheme					
m2d density	3.89	3.49	21.83	8.7E-01	1.2E+00	19.6	0.71	1.6	0.86	4.3	0.49	4.4	0.56	1.3	1.08	1.3	0.56
m2d vorticity	99.20	22.25	31.05	-1.4E+02	2.5E+01	19.6	0.71	18.4	2.14	11.8	1.21	15.5	1.29	12.9	2.22	13.8	1.49
m3d density	7.67	5.16	23.60	1.0E+00	3.0E+00	364.5	12.81	50.4	17.55	100.5	9.06	96.3	8.48	35.7	19.03	35.5	9.25
m3d pressure	27.29	23.91	31.06	-3.7E+00	2.3E+03	364.5	12.80	229.2	99.76	95.6	9.31	87.9	8.87	40.1	18.79	40.4	9.96
m3d diffusivity	36.87	23.19	30.02	0.0E+00	6.8E+00	364.5	12.68	297.6	42.90	250.8	19.09	239.3	15.02	198.8	31.92	203.0	18.47
m3d viscosity	50.07	24.86	28.59	8.6E-15	2.9E+00	364.5	12.62	314.0	46.09	249.4	18.95	246.1	14.68	209.2	32.66	207.5	19.45
h3d temp	65.70	23.54	31.56	-7.7E+01	1.0E+35	95.4	3.77	75.8	14.56	59.3	4.64	53.0	4.27	44.1	8.04	44.1	5.06
h3d pressure	81.82	24.13	31.58	-3.4E+03	1.0E+35	95.4	3.78	82.3	12.00	64.3	5.14	52.9	4.87	45.0	7.78	45.2	5.34
h3d x velocity	84.18	24.18	31.55	-5.3E+01	1.0E+35	95.4	3.89	86.1	11.27	67.4	6.22	63.3	4.59	54.5	8.86	55.4	5.44
h3d y velocity	84.32	24.18	31.55	-4.6E+01	1.0E+35	95.4	3.83	84.5	11.42	67.1	5.74	62.3	5.04	53.5	8.64	53.8	5.53
h3d z velocity	86.82	24.24	31.54	-3.2E+00	1.0E+35	95.4	3.87	88.4	10.76	85.6	8.50	76.9	5.29	68.9	9.83	69.1	6.65
M3d density	40.14	18.81	22.59	1.0E+00	3.0E+00	288.0	11.28	136.8	41.91	160.3	11.63	121.6	10.04	-	105.2	11.63	-
M3d pressure	100.00	25.17	63.00	-2.2E+00	2.2E+00	288.0	11.20	272.6	35.18	237.3	14.91	225.1	16.59	-	208.4	17.20	-
M3d x velocity	100.00	25.17	63.00	-2.2E+00	2.3E+00	288.0	10.83	275.6	32.30	230.4	14.73	215.1	15.91	-	197.7	16.84	-
M3d y velocity	100.00	25.17	63.00	-2.1E+00	2.3E+00	288.0	10.54	275.1	32.19	223.1	14.27	215.2	15.16	-	197.7	16.65	-
M3d z velocity	100.00	25.17	63.00	-5.2E+00	9.0E+00	288.0	10.32	275.5	32.62	226.6	14.74	213.7	16.05	-	196.8	16.14	-
atom x position	61.10	23.82	31.01	-4.8E-02	4.6E+02	107.7	7.07	84.3	21.18	76.0	7.88	78.8	7.61	67.3	12.88	68.6	9.07
atom y position	45.90	23.32	26.99	3.7E-02	2.1E+03	107.7	7.08	65.9	30.76	60.4	6.97	56.4	6.31	47.0	10.49	46.9	7.73
atom z position	61.68	23.84	27.48	9.1E-05	4.6E+02	107.7	7.46	94.6	19.86	82.6	9.00	86.1	8.25	75.7	13.80	75.2	9.93
atom x velocity	64.65	23.87	30.96	1.5E-01	1.4E-01	107.7	7.30	95.7	19.88	93.8	10.07	99.1	9.65	84.8	14.93	87.6	9.92
atom y velocity	64.91	23.94	27.41	3.0E-03	7.1E+03	107.7	6.69	95.7	19.76	91.6	10.27	95.9	8.34	84.6	15.02	84.6	10.31
atom energy	3.45	18.57	21.79	-3.6E+00	-2.7E+00	107.7	7.15	77.9	38.59	74.1	7.98	71.8	7.01	60.8	12.66	60.5	8.30
lucy	61.39	24.38	31.09	-6.1E+02	1.2E+03	160.5	-	137.8	-	99.5	-	90.0	-	73.6	-	77.8	-
david <sub>1mm</sub>	25.23	17.08	31.11	-4.4E+03	1.8E+03	322.5	-	144.9	-	155.7	-	163.4	-	108.6	-	131.9	-
torso	84.72	18.48	31.08	-2.7E+02	5.8E+02	1.9	-	1.7	-	1.5	-	1.5	-	1.3	-	1.3	-
rbf	71.90	20.14	25.99	1.5E+00	3.6E+02	8.4	-	7.1	-	5.8	-	5.6	-	4.7	-	4.8	-

Table 1. Compression results for the Miranda (m2d, m3d, M3d) and hurricane (h3d) structured grids, the atom point set, the lucy and david triangle meshes, and the torso and rbf tetrahedral meshes. All data but M3d is represented in single precision. The [ILS2005] scheme operates on single precision only, hence the missing values. For the meshes we report only the compressed size of vertex coordinates; timings are dominated by connectivity coding, and are hence excluded. The range measures the logarithm of the number of floating-point values between min and max. Note that the first-order entropy is limited by the number of samples in a data set.

tation.) Arguably such data sets should use an integer rather than floating-point representation, although for simplicity or other reasons it is common practice to use floating-point. Contrary to [16], which entropy codes all bits of the residual, our new coder sacrifices such potential compression gains for speed by storing these repeated low-order bits in raw and uncompressed form. However, the massive data sets from scientific simulation that motivated our work on high-speed compression, as well as our tetrahedral meshes, rarely exhibit significant low-order redundancy, as also evidenced by our results.

### 5.1.1 Lossy Compression

Fig. 3 shows that our scheme gracefully adapts to decreasing levels of precision when discarding the least significant mantissa (and eventually exponent) bits. For  $n$  bits of precision, the schemes [7, 22] require  $\log_2 n$  bits to code the number of leading zeros, whereas our scheme exploits the combination of low entropy in the leading-zero count and the elimination of the low-order bits that are most difficult to predict and compress.

### 5.2 Compression Speed

Fig. 4 shows the speed of compressing from memory to disk, including disk write time. (Because of the simplicity of our method, its decompression speed is similar to its compression speed.) We also include the raw I/O performance of dumping the data uncompressed using a single `fwrite` call. Timings

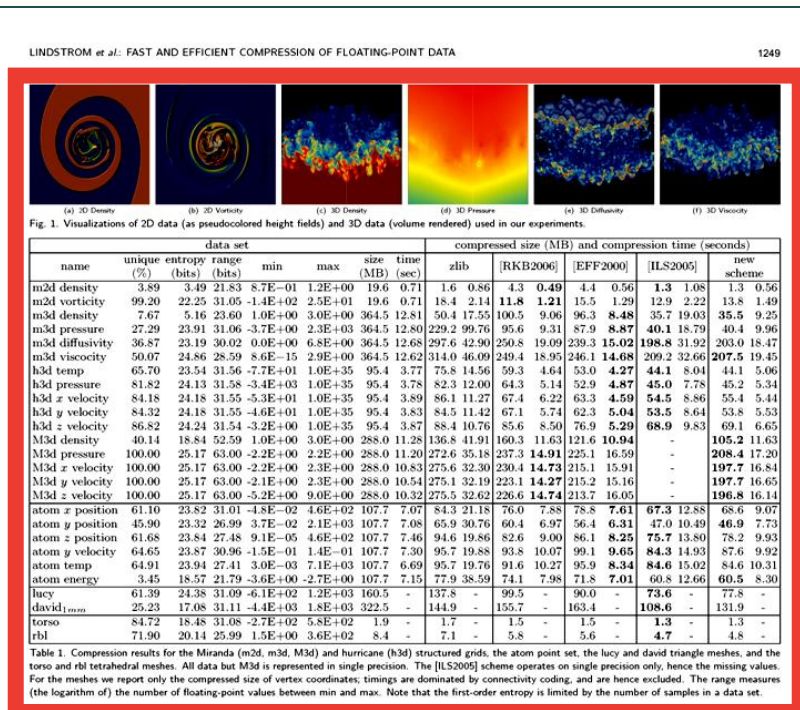
correspond to the median of five runs. Whereas our compressor is slightly slower than the less effective compressors [7, 22], it is nearly twice as fast as [16] while producing similar compression rates. However, in more I/O-intensive scenarios, such as in massively parallel simulations dumping data to the same file system (as is common), the improved compression of our method over [7, 22] results in a net gain in effective throughput. We integrated our compression code with Miranda's dump routines and ran performance tests on 256 nodes of LLNL's MCB supercomputer. Achieving on average a lossless reduction of 3.7 on 75 GB of data dumped, the overall dump time was reduced by a factor of 2.7 over writing the data uncompressed.

### 5.3 Entropy Coding

We compared the raw throughput of our range coder and Schindler's [23] by (1) passing raw bytes through it with no compression and (2) entropy coding byte sequences. In both cases, the source data was the uncompressed floating-point data used in our experiments. Timings show that our coder is 40% faster for raw transmission and 28% faster for entropy coding. Meanwhile, the inefficiency of our coder due to loss of precision and range reduction is only 26 bytes of overhead for 1.5 GB of coded data. Its raw throughput is only 20% less than an `fwrite` call, while its entropy coding throughput of 20 MB per second, which includes probability modeling and I/O time, compares favorably with state-of-the-art entropy coders [25].

# Results: evaluation scenarios (selection)

algorithmic performance  
35% of scenarios



tion.) Arguably such data sets should use an integer rather than floating-point representation, although for simplicity or other reasons it is common practice to use floating-point. Contrary to [16], which entropy codes all bits of the residual, our new coder sacrifices such potential compression gains for speed by storing these repeated low-order bits in raw and uncompressed form. However, the massive data sets from scientific simulation that motivated our work on high-speed compression, as well as our tetrahedral meshes, rarely exhibit significant low-order redundancy, as also evidenced by our results.

### 5.1.1 Lossy Compression

Fig. 3 shows that our scheme gracefully adapts to decreasing levels of precision when discarding the least significant mantissa (and eventually exponent) bits. For  $n$  bits of precision, the schemes [7, 22] require  $\log_2 n$  bits to code the number of leading zeros, whereas our scheme exploits the combination of low entropy in the leading-zero count and the elimination of the low-order bits that are most difficult to predict and compress.

### 5.2 Compression Speed

Fig. 4 shows the speed of compressing from memory to disk, including disk write time. (Because of the simplicity of our method, its decompression speed is similar to its compression speed.) We also include the raw I/O performance of dumping the data uncompressed using a single `fwrite` call. Timings

correspond to the median of five runs. Whereas our compressor is slightly slower than the less effective compressors [7, 22], it is nearly twice as fast as [16] while producing similar compression rates. However, in more I/O-intensive scenarios, such as in massively parallel simulations dumping data to the same file system (as is common), the improved compression of our method over [7, 22] results in a net gain in effective throughput. We integrated our compression code with Miranda's dump routines and ran performance tests on 256 nodes of LLNL's MCB supercomputer. Achieving on average a lossless reduction of 3.7 on 75 GB of data dumped, the overall dump time was reduced by a factor of 2.7 over writing the data uncompressed.

### 5.3 Entropy Coding

We compared the raw throughput of our range coder and Schindler's [23] by (1) passing raw bytes through it with no compression and (2) entropy coding byte sequences. In both cases, the source data was the uncompressed floating-point data used in our experiments. Timings show that our coder is 40% faster for raw transmission and 28% faster for entropy coding. Meanwhile, the inefficiency of our coder due to loss of precision and range reduction is only 26 bytes of overhead for 1.5 GB of coded data. Its raw throughput is only 20% less than an `fwrite` call, while its entropy coding throughput of 20 MB per second, which includes probability modeling and I/O time, compares favorably with state-of-the-art entropy coders [25].

# Results: evaluation scenarios (selection)

algorithmic performance  
35% of scenarios

qualitative result inspection  
46% of scenarios

LINDSTROM et al.: FAST AND EFFICIENT COMPRESSION OF FLOATING-POINT DATA 1602

Fig. 1. Visualizations of 2D data (as pseudocolored height fields) and 3D data (volume rendered) used in our experiments.

name	data set			compressed size (MB) and compression time (seconds)													
	unique (%)	entropy (bits)	range (bits)	min	max	size (MB)	time (sec)	zlib	[RKB2006]	[EFF2000]	[ILS2005]	new scheme					
m2d density	3.89	3.49	21.83	8.7E-01	1.2E+00	19.6	0.71	1.6	0.86	4.3	0.49	4.4	0.56	1.3	1.08	1.3	0.56
m2d vorticity	99.20	22.25	31.05	-1.4E+02	2.5E+01	19.6	0.71	18.4	2.14	11.8	1.21	15.5	1.29	12.9	2.22	13.8	1.49
m3d density	7.67	5.16	23.60	1.0E+00	3.0E+00	364.5	12.81	50.4	17.55	100.5	9.06	96.3	8.48	35.7	19.03	35.5	9.25
m3d pressure	27.29	23.91	31.06	-3.7E+00	2.3E+03	364.5	12.80	229.2	99.76	95.6	9.31	87.9	8.87	40.1	18.79	40.4	9.96
m3d diffusivity	36.87	23.19	30.02	0.0E+00	6.8E+00	364.5	12.68	297.6	42.90	250.8	19.09	239.3	15.02	198.8	31.92	203.0	18.47
m3d viscosity	50.07	24.86	28.59	8.6E-15	2.9E+00	364.5	12.62	314.0	46.09	249.4	18.95	246.1	14.68	209.2	32.66	207.5	19.45
h3d temp	65.70	23.54	31.56	-7.7E+01	1.0E+35	95.4	3.77	75.8	14.56	59.3	4.64	53.0	4.27	44.1	8.04	44.1	5.06
h3d pressure	81.82	24.13	31.58	-3.4E+03	1.0E+35	95.4	3.78	82.3	12.00	64.3	5.14	52.9	4.87	45.0	7.78	45.2	5.34
h3d x velocity	84.18	24.18	31.55	-5.3E+01	1.0E+35	95.4	3.89	86.1	11.27	67.4	6.22	63.3	4.59	54.5	8.86	55.4	5.44
h3d y velocity	84.32	24.18	31.55	-4.6E+01	1.0E+35	95.4	3.83	84.5	11.42	67.1	5.74	62.3	5.04	53.5	8.64	53.8	5.53
h3d z velocity	86.82	24.24	31.54	-3.2E+00	1.0E+35	95.4	3.87	88.4	10.76	85.6	8.50	76.9	5.29	68.9	9.83	69.1	6.65
M3d density	40.14	18.81	22.59	1.0E+00	3.0E+00	288.0	11.28	136.8	41.91	160.5	11.63	121.6	10.04	-	-	105.2	11.63
M3d pressure	100.00	25.17	63.00	-2.2E+00	2.2E+00	288.0	11.20	272.6	35.18	237.3	14.91	225.1	16.59	-	-	208.4	17.20
M3d x velocity	100.00	25.17	63.00	-2.2E+00	2.3E+00	288.0	10.83	275.6	32.30	230.4	14.73	215.1	15.91	-	-	197.7	16.84
M3d y velocity	100.00	25.17	63.00	-2.1E+00	2.3E+00	288.0	10.54	275.1	32.19	223.1	14.27	215.2	15.16	-	-	197.7	16.65
M3d z velocity	100.00	25.17	63.00	-5.2E+00	9.0E+00	288.0	10.32	275.5	32.62	226.6	14.74	213.7	16.05	-	-	196.8	16.14
atom x position	61.10	23.82	31.01	-4.8E-02	4.6E+02	107.7	7.07	84.3	21.18	76.0	7.88	78.8	7.61	67.3	12.88	68.6	9.07
atom y position	45.90	23.32	26.99	3.7E-02	2.1E+03	107.7	7.08	65.9	30.76	60.4	6.97	56.4	6.31	47.0	10.49	46.9	7.73
atom z position	61.68	23.84	27.48	9.1E-05	4.6E+02	107.7	7.46	94.6	19.86	82.6	9.00	86.1	8.25	75.7	13.80	78.2	9.93
atom y velocity	64.65	23.87	30.96	1.5E-01	1.4E-01	107.7	7.30	95.7	19.88	93.8	10.07	99.1	9.65	84.8	14.93	87.6	9.92
atom temp	64.91	23.94	27.41	3.0E-03	7.1E+03	107.7	6.69	91.6	10.27	95.9	8.34	84.6	15.02	84.6	15.02	84.6	10.31
atom energy	3.45	18.57	21.79	-3.6E+00	-2.7E+00	107.7	7.15	77.9	38.59	74.1	7.98	71.8	7.01	60.8	12.66	60.5	8.30
lucy	61.39	24.38	31.09	-6.1E+02	1.2E+03	160.5	-	137.8	-	99.5	-	90.0	-	73.6	-	77.8	-
david1mm	25.23	17.08	31.11	-4.4E+03	1.8E+03	322.5	-	144.9	-	155.7	-	163.4	-	108.6	-	131.9	-
torso	84.72	18.48	31.08	-2.7E+02	5.8E+02	1.9	-	1.7	-	1.5	-	1.5	-	1.3	-	1.3	-
rhl	71.90	20.14	25.99	1.5E+00	3.6E+02	8.4	-	7.1	-	5.8	-	5.6	-	4.7	-	4.8	-

Table 1. Compression results for the Miranda (m2d, m3d, M3d) and hurricane (h3d) structured grids, the atom point set, the lucy and david triangle meshes, and the torso and rhl tetrahedral meshes. All data but M3d is represented in single precision. The [ILS2005] scheme operates on single precision only, hence the missing values. For the meshes we report only the compressed size of vertex coordinates; timings are dominated by connectivity coding, and are hence excluded. The range measures the (logarithm of) the number of floating-point values between min and max. Note that the first-order entropy is limited by the number of samples in a data set.

5.1.1 Lossy Compression

Fig. 3 shows that our scheme gracefully adapts to decreasing levels of precision when discarding the least significant mantissa (and eventually exponent) bits. For  $n$  bits of precision, the schemes [7, 22] require  $\log_2 n$  bits to code the number of leading zeros, whereas our scheme exploits the combination of low entropy in the leading-zero count and the elimination of the low-order bits that are most difficult to predict and compress.

5.2 Compression Speed

Fig. 4 shows the speed of compressing from memory to disk, including disk write time. (Because of the small size of our method, its decompression speed is similar to its compression speed.) We also include the raw I/O performance of dumping the data uncompressed using a single `fwrite` call. Timings

correspond to the median of five runs. Whereas our compressor is slightly slower than the less effective compressors [7, 22], it is nearly twice as fast as [16] while producing similar compression rates. However, in more I/O-intensive scenarios, such as in massively parallel simulations dumping data to the same file system (as is common), the improved compression of our method over [7, 22] results in a net gain in effective throughput. We integrated our compression code with Miranda's dump routines and ran performance tests on 276 nodes of LLNL's MCH supercomputer. Achieving on average a lossless reduction of 3.7 on 75 GB of data dumped, the overall dump time was reduced by a factor of 2.7 over writing the data uncompressed.

5.3 Entropy Coding

We compared the raw throughput of our range coder and Schindler's [23] by (1) passing raw bytes through it with no compression and (2) entropy coding byte sequences. In both cases, the source data was the uncompressed floating-point data used in our experiments. Timings show that our coder is 40% faster for raw transmission and 28% faster for entropy coding. Meanwhile, the inefficiency of our coder due to loss of precision and range reduction is only 26 bytes of overhead for 3.5 GB of coded data. Its raw throughput is only 20% less than an `fwrite` call, while its entropy coding throughput of 20 MB per second, which includes probability modeling and I/O time, compares favorably with state-of-the-art entropy coders [25].

Lindstrom et al. [2006]

1602 IEEE TRANSACTIONS ON VISUALIZATION AND COMPUTER GRAPHICS, VOL. 16, NO. 6, NOVEMBER/DECEMBER 2010

Fig. 8. Glyphs in the double point load stress tensor field reveal the minor eigenvector along which hyperstreamlines [7] are traced (a), and the variation in stress with distance from the load (b).

Fig. 9. Visualization of geometry tensors (a) and Hessians associated with isosurfaces (b) and two different definitions of edges, zero-crossings of the second-directional derivative (c) and the Laplacian (d). These results use  $s(\|D\|) \propto \|D\|^{1/2}$  in (6).

Fig. 10(b) uses our superquadratic glyphs with  $s(\|D\|) \propto \|D\|$ . The aspect ratio reflects the relative eigenvalue magnitudes, the size correctly indicates the tensor norm, and pointed glyphs shapes clearly communicate eigenvector directions. With compression of scale variation

volume rendered [32], but its eigenvectors are commonly used in non-photo-realistic rendering, e.g. curvature-based strokes [11, 14, 19]. Inspecting geometry tensors could help debug an NPR method giving unexpected results in an unfamiliar dataset. Fig. 9(a) visualizes geometry tensors  $G$  on an isosurface (sampled by a particle system [39]) of an ear from the Visible Human male CT scan. Variations in surface curvature are reflected in the new glyphs: convex (blue circles), concave (orange circles), and saddles (orange and blue stars). For comparison, Fig. 9(b) shows the full Hessian  $H$  from which  $G$  was computed.

The new glyphs may also have a role in visualizing the tensor ingredients of image analysis methods such as edge detection. One edge definition is zero-crossing on the second directional derivative along the gradient direction,  $f'' = \mathbf{n}^T H \mathbf{n}$ . This edge surface is sampled by a particle system [33] in Fig. 9(c), showing the Hessians at the edge locations, and revealing close similarities with the geometry tensors on the isosurface in Fig. 9(a), indicating that one of the Hessian eigenvalues is near zero even though this is not part of the edge definition. Another edge definition is the zero-crossing of the Laplacian  $\nabla^2 f = \text{tr}(H)$ , and Fig. 9(d) illustrates the difference between the Hessians on this surface and those in Fig. 9(c). The consistently gray glyph halos in Fig. 9(d) indicate that these are traceless tensors.

As a demonstration of the glyphs in a 2-D visualization, Fig. 10 visualizes a cross-section of a simulation of jet flow rightward into a steady medium, causing turbulence. Glyphs of rate-of-deformations tensors document how an infinitesimal volume is stretched or compressed as it moves along the flow. A backdrop of line integral convolution [4] (with contrast modulated by velocity) provides visual context. Fig. 10(a) uses the exponentially-scaled ellipses of [34] to map tensors with negative eigenvalues to positive-definite tensors suitable for ellipsoid visualization. When the absolute difference between eigenvalues becomes too large, these glyphs can become so stretched that they overlap each other and extend over a significant portion of the domain, undermining the locality normally enjoyed by glyphs. Such stretching also reduces the visual presence of the needle-like glyphs for tensors with larger norms, contrary to scale preservation (6). Fig. 10(b) uses our superquadratic glyphs with  $s(\|D\|) \propto \|D\|$ . The aspect ratio reflects the relative eigenvalue magnitudes, the size correctly indicates the tensor norm, and pointed glyphs shapes clearly communicate eigenvector directions. With compression of scale variation

6 CONCLUSION

Visualization research has made significant progress in visualizing second-order tensor fields, but has mostly concentrated on the positive-definite case. Faced with indefinite tensors, a frequent strategy is to map them to positive-definite tensors prior to visualization [34, 22, 21, 52, 33]. Even when bijective mappings are used (so mathematically, no information is lost), such mappings still visually obscure the difference between positive and negative eigenvalues, which is a fundamental qualitative aspect in various applications.

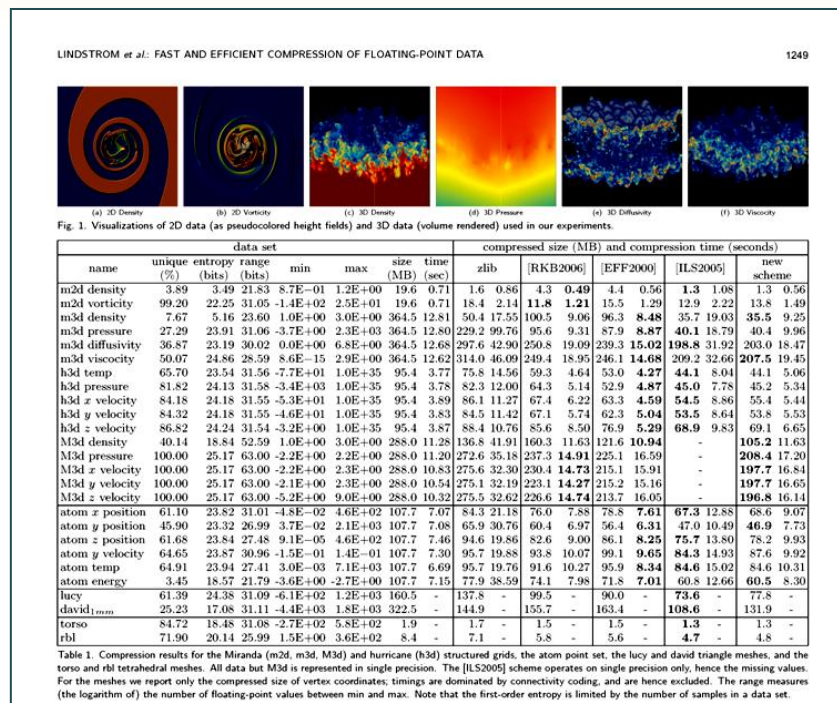
Therefore, we propose an extension of a previous positive-definite tensor glyph [28] to the full space of symmetric second-order tensors. Our glyph emphasizes differences in eigenvalue sign in a way that, unlike the Reynolds glyph [18], prevents small eigenvalues from being occluded by larger ones. We also propose to use halos to ensure tensor glyph visibility even when one or more eigenvalues are near zero. Finally, we present a time- and memory-efficient implementa-

Schultz & Kindlmann [2010]

# Results: evaluation scenarios (selection)

algorithmic performance  
35% of scenarios

qualitative result inspection  
46% of scenarios



ation.) Arguably such data sets should use an integer rather than floating-point representation, although for simplicity or other reasons it is common practice to use floating-point. Contrary to [16], which entropy codes all bits of the residual, our new coder sacrifices such potential compression gains for speed by storing these repeated low-order bits in raw and uncompressed form. However, the massive data sets from scientific simulation that motivated our work on high-speed compression, as well as our tetrahedral meshes, rarely exhibit significant low-order redundancy, as also evidenced by our results.

### 5.1.1 Lossy Compression

Fig. 3 shows that our scheme gracefully adapts to decreasing levels of precision when discarding the least significant mantissa (and eventually exponent) bits. For  $n$  bits of precision, the schemes [7, 22] require  $\log_2 n$  bits to code the number of leading zeros, whereas our scheme exploits the combination of low entropy in the leading-zero count and the elimination of the low-order bits that are most difficult to predict and compress.

### 5.2 Compression Speed

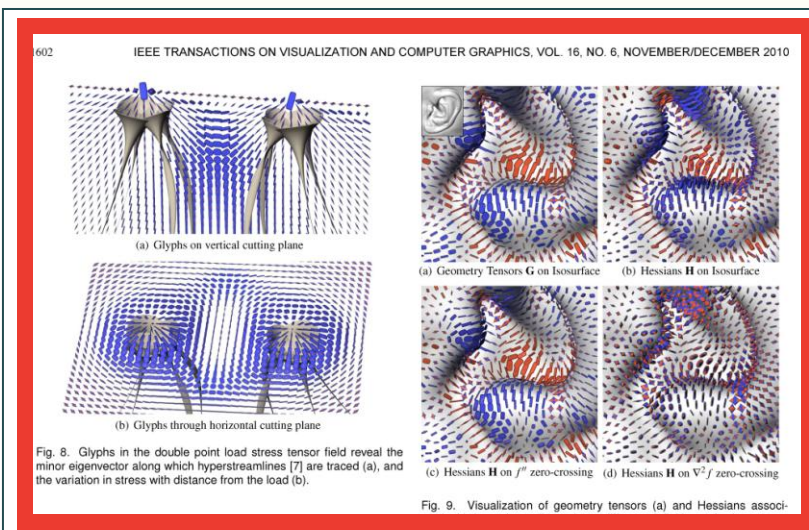
Fig. 4 shows the speed of compressing from memory to disk, including disk write time. (Because of the small size of our test method, its decompression speed is similar to its compression speed.) We also include the raw I/O performance of dumping the data uncompressed using a single `fwrite` call. Timings

correspond to the median of five runs. Whereas our compressor is slightly slower than the less effective compressors [7, 22], it is nearly twice as fast as [16] while producing similar compression rates. However, in more I/O-intensive scenarios, such as in massively parallel simulations dumping data to the same file system (as is common), the improved compression of our method over [7, 22] results in a net gain in effective throughput. We integrated our compression code with Miranda's dump routines and ran performance tests on 276 nodes of LLNL's MCH supercomputer. Achieving on average a lossless reduction of 3.7 on 75 GB of data dumped, the overall dump time was reduced by a factor of 2.7 over writing the data uncompressed.

### 5.3 Entropy Coding

We compared the raw throughput of our range coder and Schindler's [23] by (1) passing raw bytes through it with no compression and (2) entropy coding byte sequences. In both cases, the source data was the uncompressed floating-point data used in our experiments. Timings show that our coder is 40% faster for raw transmission and 28% faster for entropy coding. Meanwhile, the inefficiency of our coder due to loss of precision and range reduction is only 26 bytes of overhead for 3.5 GB of coded data. Its raw throughput is only 20% less than an `fwrite` call, while its entropy coding throughput of 20 MB per second, which includes probability modeling and I/O time, compares favorably with state-of-the-art entropy coders [25].

Lindstrom et al. [2006]



volume rendered [32], but its eigenvectors are commonly used in non-photo-realistic rendering, e.g. curvature-based strokes [11, 14, 19]. Inspecting geometry tensors could help debug an NPR method giving unexpected results in an unfamiliar dataset. Fig. 9(a) visualizes geometry tensors  $G$  on an isosurface (sampled by a particle system [39]) of an ear from the Visible Human male CT scan. Variations in surface curvature are reflected in the new glyphs: convex (blue circles), concave (orange circles), and saddles (orange and blue stars). For comparison, Fig. 9(b) shows the full Hessian  $H$  from which  $G$  was computed.

The new glyphs may also have a role in visualizing the tensor ingredients of image analysis methods such as edge detection. One edge definition is zero-crossing on the second directional derivative along the gradient direction,  $f'' = \mathbf{n}^T H \mathbf{n}$ . This edge surface is sampled by a particle system [33] in Fig. 9(c), showing the Hessians at the edge locations, and revealing close similarities with the geometry tensors on the isosurface in Fig. 9(a), indicating that one of the Hessian eigenvalues is near zero even though this is not part of the edge definition. Another edge definition is the zero-crossing of the Laplacian  $\nabla^2 f = \text{tr}(H)$ , and Fig. 9(d) illustrates the difference between the Hessians on this surface and those in Fig. 9(c). The consistently gray glyph halos in Fig. 9(d) indicate that these are traceless tensors.

As a demonstration of the glyphs in a 2-D visualization, Fig. 10 visualizes a cross-section of a simulation of jet flow rightward into a steady medium, causing turbulence. Glyphs of rate-of-deformations tensors document how an infinitesimal volume is stretched or compressed as it moves along the flow. A backdrop of line integral convolution [4] (with contrast modulated by velocity) provides visual context. Fig. 10(a) uses the exponentially-scaled ellipses of [34] to map tensors with negative eigenvalues to positive-definite tensors suitable for ellipsoid visualization. When the absolute difference between eigenvalues becomes too large, these glyphs can become so stretched that they overlap each other and extend over a significant portion of the domain, undermining the locality normally enjoyed by glyphs. Such stretching also reduces the visual presence of the needle-like glyphs for tensors with larger norms, contrary to scale preservation (6). Fig. 10(b) uses our superquadratic glyphs with  $s(\|D\|) \propto \|D\|$ . The aspect ratio reflects the relative eigenvalue magnitudes, the size correctly indicates the tensor norm, and pointed glyph shapes clearly communicate eigenvector directions. With compression of scale variation

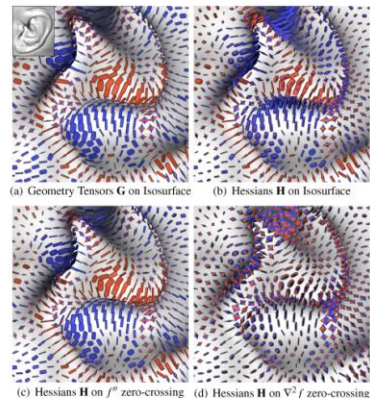


Fig. 9. Visualization of geometry tensors (a) and Hessians associated with the second-order directional derivative (c) and the Laplacian (d). These results use  $s(\|D\|) \propto \|D\|^{1/2}$  in (6).

$s(\|D\|) \propto \|D\|^{1/2}$ . Fig. 10(c) better shows the directional patterns where the tensor norm is low. Colormapping the rate-of-deformation tensor trace with glyph halos highlights the regions of over-all stretching or compression, especially along the bottom edge of the domain.

Finally, Fig. 11 demonstrates how our new glyph performs traceless tensor visualization, in a side-by-side comparison to the dedicated traceless NLG tensor glyphs by Jankun-Kelly et al. [25]. Traceless tensors form a plane in eigenvalue space, and we are visualizing samples from a square within this plane, centered around the zero tensor (cf. Fig. 4(c)). Unlike the traceless glyph, which maps tensor norm to glyph sharpness, our glyph expresses norm by its overall scale  $s(\|D\|) \propto \|D\|$ . Consequently, the traceless glyph requires pre-specification of maximum eigenvalue magnitudes (which are mapped to perfect sharpness), while our glyph can be used without such prior information. Another notable difference is that limiting their glyph to traceless tensors allows Jankun-Kelly et al. to make use of parts of the superquadratic shape space – including cylinders and boxes – that our approach sets aside for positive- or negative-definite tensors.

### 6 CONCLUSION

Visualization research has made significant progress in visualizing second-order tensor fields, but has mostly concentrated on the positive-definite case. Faced with indefinite tensors, a frequent strategy is to map them to positive-definite tensors prior to visualization [34, 22, 21, 52, 33]. Even when bijective mappings are used (so mathematically, no information is lost), such mappings still visually obscure the difference between positive and negative eigenvalues, which is a fundamental qualitative aspect in various applications.

Therefore, we propose an extension of a previous positive-definite tensor glyph [28] to the full space of symmetric second-order tensors. Our glyph emphasizes differences in eigenvalue sign in a way that, unlike the Reynolds glyph [18], prevents small eigenvalues from being occluded by larger ones. We also propose to use halos to ensure tensor glyph visibility even when one or more eigenvalues are near zero. Finally, we present a time- and memory-efficient implementa-

Schultz & Kindlmann [2010]

# Results: evaluation scenarios (selection)

algorithmic performance  
35% of scenarios

qualitative result inspection  
46% of scenarios

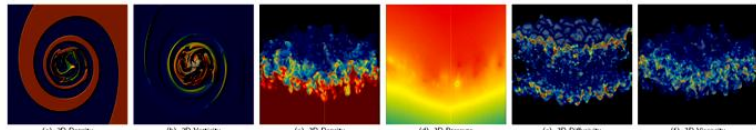


Fig. 1. Visualizations of 2D data (as pseudocolored height fields) and 3D data (volume rendered) used in our experiments.

name	data set				compressed size (MB) and compression time (seconds)												
	unique (%)	entropy (bits)	range (bits)	min max	size (MB)	time (sec)	zlib	[RKB2006]	[EFF2000]	[ILS2005]	new scheme						
m2d density	3.89	3.49	21.83	8.7E-01	1.2E+00	19.6	0.71	1.6	0.86	4.3	0.49	4.4	0.56	1.3	1.08	1.3	0.56
m2d vorticity	99.20	22.25	31.05	-1.4E+02	2.5E+01	19.6	0.71	18.4	2.14	11.8	1.21	15.5	1.29	12.9	2.22	13.8	1.49
m3d density	7.67	5.16	23.60	1.0E+00	3.0E+00	364.5	12.81	50.4	17.55	100.5	9.06	96.3	8.48	35.7	19.03	35.5	9.25
m3d pressure	27.29	23.91	31.06	-3.7E+00	2.3E+03	364.5	12.80	229.2	99.76	95.6	9.31	87.9	8.87	40.1	18.79	40.4	9.96
m3d diffusivity	36.87	23.19	30.02	0.0E+00	6.8E+00	364.5	12.68	297.6	42.90	250.8	19.09	239.3	15.02	198.8	31.92	203.0	18.47
m3d viscosity	50.07	24.86	28.59	8.6E-15	2.9E+00	364.5	12.62	314.0	46.09	249.4	18.95	246.1	14.68	209.2	32.66	207.5	19.45
h3d temp	65.70	23.54	31.56	-7.7E+01	1.0E+35	95.4	3.77	75.8	14.56	59.3	4.64	53.0	4.27	44.1	8.04	44.1	5.06
h3d pressure	81.82	24.13	31.58	-3.4E+03	1.0E+35	95.4	3.78	82.3	12.00	64.3	5.14	52.9	4.87	45.0	7.78	45.2	5.34
h3d x velocity	84.18	24.18	31.55	-5.3E+01	1.0E+35	95.4	3.89	86.1	11.27	67.4	6.22	63.3	4.59	54.5	8.86	55.4	5.44
h3d y velocity	84.32	24.18	31.55	-4.6E+01	1.0E+35	95.4	3.83	84.5	11.42	67.1	5.74	62.3	5.04	53.5	8.64	53.8	5.53
h3d z velocity	86.82	24.24	31.54	-3.2E+00	1.0E+35	95.4	3.87	88.4	10.76	85.6	8.50	76.9	5.29	68.9	9.83	69.1	6.65
M3d density	40.14	18.81	22.59	1.0E+00	3.0E+00	288.0	11.28	136.8	41.91	160.5	11.63	121.6	10.94	-	-	105.2	11.63
M3d pressure	100.00	25.17	63.00	-2.2E+00	2.2E+00	288.0	11.20	272.6	35.18	237.3	14.91	225.1	16.59	-	-	208.4	17.20
M3d x velocity	100.00	25.17	63.00	-2.2E+00	2.3E+00	288.0	10.83	275.6	32.30	230.4	14.73	215.1	15.91	-	-	197.7	16.84
M3d y velocity	100.00	25.17	63.00	-2.1E+00	2.3E+00	288.0	10.54	275.1	32.19	223.1	14.27	215.2	15.16	-	-	197.7	16.65
M3d z velocity	100.00	25.17	63.00	-5.2E+00	9.0E+00	288.0	10.32	275.5	32.62	226.6	14.74	213.7	16.05	-	-	196.8	16.14
atom x position	61.10	23.82	31.01	-4.8E-02	4.6E+02	107.7	7.07	84.3	21.18	76.0	7.88	78.8	7.61	67.3	12.88	68.6	9.07
atom y position	45.90	23.32	26.99	3.7E-02	2.1E+03	107.7	7.08	65.9	30.76	60.4	6.97	56.4	6.31	47.0	10.49	46.9	7.73
atom z position	61.68	23.84	27.48	9.1E-05	4.6E+02	107.7	7.46	94.6	19.86	82.6	9.00	86.1	8.25	75.7	13.80	78.2	9.93
atom x velocity	64.65	23.87	30.96	1.5E-01	1.4E-01	107.7	7.30	95.7	19.88	93.8	10.07	99.1	9.65	84.3	14.93	87.6	9.92
atom y velocity	64.91	23.94	27.41	3.0E-03	7.1E+03	107.7	6.69	91.6	10.27	95.9	8.34	84.6	15.02	84.6	15.02	84.6	10.31
atom energy	3.45	18.57	21.79	-3.6E+00	-2.7E+00	107.7	7.15	77.9	38.59	74.1	7.98	71.8	7.01	60.8	12.66	60.5	8.30
lucy	61.39	24.38	31.09	-6.1E+02	1.2E+03	160.5	-	137.8	-	99.5	-	90.0	-	73.6	-	77.8	-
david1mm	25.23	17.08	31.11	-4.4E+03	1.8E+03	322.5	-	144.9	-	155.7	-	163.4	-	108.6	-	131.9	-
torso	84.72	18.48	31.08	-2.7E+02	5.8E+02	1.9	-	1.7	-	1.5	-	1.5	-	1.3	-	1.3	-
rhl	71.90	20.14	25.99	1.5E+00	3.6E+02	8.4	-	7.1	-	5.8	-	5.6	-	4.7	-	4.8	-

Table 1. Compression results for the Miranda (m2d, m3d, M3d) and hurricane (h3d) structured grids, the atom point set, the lucy and david triangle meshes, and the torso and rhl tetrahedral meshes. All data but M3d is represented in single precision. The [ILS2005] scheme operates on single precision only, hence the missing values. For the meshes we report only the compressed size of vertex coordinates; timings are dominated by connectivity coding, and are hence excluded. The range measures the logarithm of the number of floating-point values between min and max. Note that the first-order entropy is limited by the number of samples in a data set.

ation.) Arguably such data sets should use an integer rather than floating-point representation, although for simplicity or other reasons it is common practice to use floating-point. Contrary to [16], which entropy codes all bits of the residual, our new coder sacrifices such potential compression gains for speed by storing these repeated low-order bits in raw and uncompressed form. However, the massive data sets from scientific simulation that motivated our work on high-speed compression, as well as our tetrahedral meshes, rarely exhibit significant low-order redundancy, as also evidenced by our results.

### 5.1.1 Lossy Compression

Fig. 3 shows that our scheme gracefully adapts to decreasing levels of precision when discarding the least significant mantissa (and eventually exponent) bits. For  $n$  bits of precision, the schemes [7, 22] require  $\log_2 n$  bits to code the number of leading zeros, whereas our scheme exploits the combination of low entropy in the leading-zero count and the elimination of the low-order bits that are most difficult to predict and compress.

### 5.2 Compression Speed

Fig. 4 shows the speed of compressing from memory to disk, including disk write time. (Because of the slow disk write method, its decompression speed is similar to its compression speed.) We also include the raw I/O performance of dumping the data uncompressed using a single `fwrite` call. Timings

correspond to the median of five runs. Whereas our compressor is slightly slower than the less effective compressors [7, 22], it is nearly twice as fast as [16] while producing similar compression rates. However, in more I/O-intensive scenarios, such as in massively parallel simulations dumping data to the same file system (as is common), the improved compression of our method over [7, 22] results in a net gain in effective throughput. We integrated our compression code with Miranda's dump routines and ran performance tests on 276 nodes of LLNL's MCH supercomputer. Achieving on average a lossless reduction of 3.7 on 75 GB of data dumped, the overall dump time was reduced by a factor of 2.7 over writing the data uncompressed.

### 5.3 Entropy Coding

We compared the raw throughput of our range coder and Schindler's [23] by (1) passing raw bytes through it with no compression and (2) entropy coding byte sequences. In both cases, the source data was the uncompressed floating-point data used in our experiments. Timings show that our coder is 40% faster for raw transmission and 28% faster for entropy coding. Meanwhile, the inefficiency of our coder due to loss of precision and range reduction is only 26 bytes of overhead for 3.5 GB of coded data. Its raw throughput is only 20% less than an `fwrite` call, while its entropy coding throughput of 20 MB per second, which includes probability modeling and I/O time, compares favorably with state-of-the-art entropy coders [25].

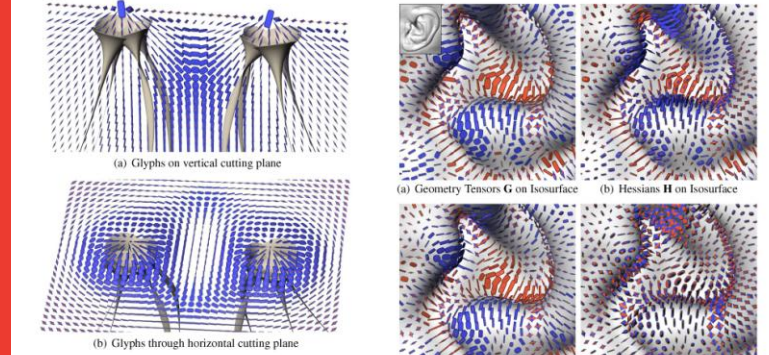


Fig. 8. Glyphs in the double point load stress tensor field reveal the minor eigenvector along which hyperstreamlines [7] are traced (a), and the variation in stress with distance from the load (b).

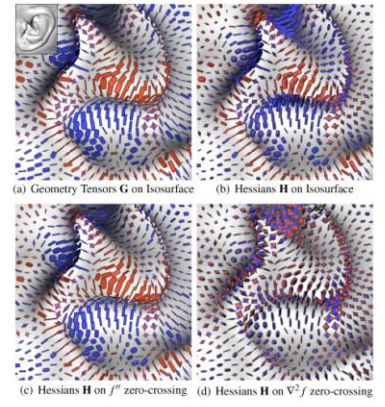


Fig. 9. Visualization of geometry tensors (a) and Hessians associated with the second-order directional derivative (c) and the Laplacian (d). These results use  $s(\|\mathbf{D}\|) \propto \|\mathbf{D}\|^{1/2}$  in (6).

volume rendered [32], but its eigenvectors are commonly used in non-photo-realistic rendering, e.g. curvature-based strokes [11, 14, 19]. Inspecting geometry tensors could help debug an NPR method giving unexpected results in an unfamiliar dataset. Fig. 9(a) visualizes geometry tensors  $G$  on an isosurface (sampled by a particle system [39]) of an ear from the Visible Human male CT scan. Variations in surface curvature are reflected in the new glyphs: convex (blue circles), concave (orange circles), and saddles (orange and blue stars). For comparison, Fig. 9(b) shows the full Hessian  $H$  from which  $G$  was computed.

The new glyphs may also have a role in visualizing the tensor invariants of image analysis methods such as edge detection. One edge definition is zero-crossing on the second directional derivative along the gradient direction,  $f'' = \mathbf{n}^T H \mathbf{n}$ . This edge surface is sampled by a particle system [33] in Fig. 9(c), showing the Hessians at the edge locations, and revealing close similarities with the geometry tensors on the isosurface in Fig. 9(a), indicating that one of the Hessian eigenvalues is near zero even though this is not part of the edge definition. Another edge definition is the zero-crossing of the Laplacian  $\nabla^2 f = \text{tr}(H)$ , and Fig. 9(d) illustrates the difference between the Hessians on this surface and those in Fig. 9(c). The consistently gray glyph halos in Fig. 9(d) indicate that these are traceless tensors.

As a demonstration of the glyphs in a 2-D visualization, Fig. 10 visualizes a cross-section of a simulation of jet flow rightward into a steady medium, causing turbulence. Glyphs of rate-of-deformations tensors document how an infinitesimal volume is stretched or compressed as it moves along the flow. A backdrop of line integral convolution [4] (with contrast modulated by velocity) provides visual context. Fig. 10(a) uses the exponentially-scaled ellipses of [34] to map tensors with negative eigenvalues to positive-definite tensors suitable for ellipsoid visualization. When the absolute difference between eigenvalues becomes too large, these glyphs can become so stretched that they overlap each other and extend over a significant portion of the domain, undermining the locality normally enjoyed by glyphs. Such stretching also reduces the visual presence of the needle-like glyphs for tensors with larger norms, contrary to scale preservation (6). Fig. 10(b) uses our superquadratic glyphs with  $s(\|\mathbf{D}\|) \propto \|\mathbf{D}\|$ . The aspect ratio reflects the relative eigenvalue magnitudes, the size correctly indicates the tensor norm, and pointed glyph shapes clearly communicate eigenvector directions. With compression of scale variation

crossings of the second-order directional derivative (c) and the Laplacian (d). These results use  $s(\|\mathbf{D}\|) \propto \|\mathbf{D}\|^{1/2}$  in (6). Fig. 10(c) better shows the directional patterns where the tensor norm is low. Colormapping the rate-of-deformation tensor trace with glyph halos highlights the regions of over-all stretching or compression, especially along the bottom edge of the domain.

Finally, Fig. 11 demonstrates how our new glyph performs traceless tensor visualization, in a side-by-side comparison to the dedicated traceless NLG tensor glyphs by Jankun-Kelly et al. [25]. Traceless tensors form a plane in eigenvalue space, and we are visualizing samples from a square within this plane, centered around the zero tensor (cf. Fig. 4(c)). Unlike the traceless glyph, which maps tensor norm to glyph sharpness, our glyph expresses norm by its overall scale  $s(\|\mathbf{D}\|) \propto \|\mathbf{D}\|$ . Consequently, the traceless glyph requires pre-specification of maximum eigenvalue magnitudes (which are mapped to perfect sharpness), while our glyph can be used without such prior information. Another notable difference is that limiting their glyph to traceless tensors allows Jankun-Kelly et al. to make use of parts of the superquadratic shape space – including cylinders and boxes – that our approach sets aside for positive- or negative-definite tensors.

### 6 CONCLUSION

Visualization research has made significant progress in visualizing second-order tensor fields, but has mostly concentrated on the positive-definite case. Faced with indefinite tensors, a frequent strategy is to map them to positive-definite tensors prior to visualization [34, 22, 21, 52, 33]. Even when bijective mappings are used (so mathematically, no information is lost), such mappings still visually obscure the difference between positive and negative eigenvalues, which is a fundamental qualitative aspect in various applications.

Therefore, we propose an extension of a previous positive-definite tensor glyph [28] to the full space of symmetric second-order tensors. Our glyph emphasizes differences in eigenvalue sign in a way that, unlike the Reynolds glyph [18], prevents small eigenvalues from being occluded by larger ones. We also propose to use halos to ensure tensor glyph visibility even when one or more eigenvalues are near zero. Finally, we present a time- and memory-efficient implementa-

Linstrom et al. [2006]

Schultz & Kindlmann [2010]

# Results: evaluation scenarios (selection)

algorithmic performance  
35% of scenarios

qualitative result inspection  
46% of scenarios

user performance/experience  
14% of scenarios

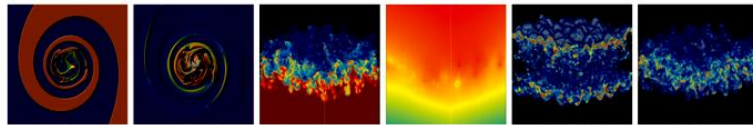


Fig. 1. Visualizations of 2D data (as pseudocolored height fields) and 3D data (volume rendered) used in our experiments.

data set	unique entropy range				compressed size (MB) and compression time (seconds)												
	name	(%)	(bits)	(min)	size (MB)	time (sec)	zlib	[RKB2006]	[EFF2000]	[LS2005]	new scheme						
m2d density	3.89	3.49	21.83	8.7E-01	1.2E+00	19.6	0.71	1.6	0.86	4.3	0.49	4.4	0.56	1.3	1.08	1.3	0.56
m2d vorticity	99.20	22.25	31.05	-1.4E+02	2.5E+01	19.6	0.71	18.4	2.14	11.8	1.21	15.5	1.29	12.9	2.22	13.8	1.49
m3d density	7.67	5.16	23.60	1.0E+00	3.0E+00	364.5	12.81	50.4	17.55	100.5	9.06	96.3	8.48	35.7	19.03	35.5	9.25
m3d pressure	27.29	23.91	31.06	-3.7E-00	2.3E+03	364.5	12.80	229.2	99.76	95.6	9.31	87.9	8.87	40.1	18.79	40.4	9.96
m3d diffusivity	36.87	23.19	30.02	0.0E+00	6.8E+00	364.5	12.68	297.6	42.90	250.8	19.09	239.3	15.02	198.8	31.92	203.0	18.47
m3d viscosity	50.07	24.86	28.59	8.6E-15	2.9E+00	364.5	12.62	314.0	46.09	249.4	18.95	246.1	14.68	209.2	32.66	207.5	19.45
h3d temp	65.70	23.54	31.56	-7.7E+01	1.0E+35	95.4	3.77	75.8	14.56	59.3	4.64	53.0	4.27	44.1	8.04	44.1	5.06
h3d pressure	81.82	24.13	31.58	-3.4E+03	1.0E+35	95.4	3.78	82.3	12.00	64.3	5.14	52.9	4.87	45.0	7.78	45.2	5.34
h3d x velocity	84.18	24.18	31.55	-5.3E+01	1.0E+35	95.4	3.89	86.1	11.27	67.4	6.22	63.3	4.59	54.5	8.86	55.4	5.44
h3d y velocity	84.32	24.18	31.55	-4.6E+01	1.0E+35	95.4	3.83	84.5	11.42	67.1	5.74	62.3	5.04	53.5	8.64	53.8	5.53
h3d z velocity	86.82	24.24	31.54	-3.2E+00	1.0E+35	95.4	3.87	88.4	10.76	85.6	8.50	76.9	5.29	68.9	9.83	69.1	6.65
M3d density	40.14	18.81	22.59	1.0E+00	3.0E+00	288.0	11.28	136.8	41.91	160.3	11.63	121.6	10.94	-	-	105.2	11.63
M3d pressure	100.00	25.17	63.00	-2.2E+00	2.2E+00	288.0	11.20	272.6	35.18	237.3	14.91	225.1	16.59	-	-	208.4	17.20
M3d x velocity	100.00	25.17	63.00	-2.2E+00	2.3E+00	288.0	10.83	275.6	32.30	230.4	14.73	215.1	15.91	-	-	197.7	16.84
M3d y velocity	100.00	25.17	63.00	-2.1E+00	2.3E+00	288.0	10.54	275.1	32.19	223.1	14.27	215.2	15.16	-	-	197.7	16.65
M3d z velocity	100.00	25.17	63.00	-5.2E+00	9.0E+00	288.0	10.32	275.5	32.62	226.6	14.74	213.7	16.05	-	-	196.8	16.14
atom x position	61.10	23.82	31.01	-4.8E-02	4.6E+02	107.7	7.07	84.3	21.18	76.0	7.88	78.8	7.61	67.3	12.88	68.6	9.07
atom y position	45.90	23.32	26.99	3.7E-02	2.1E+03	107.7	7.08	65.9	30.76	60.4	6.97	56.4	6.31	47.0	10.49	46.9	7.73
atom z position	61.68	23.84	27.48	9.1E-05	4.6E+02	107.7	7.46	94.6	19.86	82.6	9.00	86.1	8.25	75.7	13.80	78.2	9.93
atom x velocity	64.65	23.87	30.96	1.5E-01	1.4E-01	107.7	7.30	95.7	19.88	93.8	10.07	99.1	9.65	84.8	14.93	87.6	9.92
atom temp	64.91	23.94	27.41	3.0E-03	7.1E+03	107.7	6.69	95.7	19.76	91.6	10.27	95.9	8.34	84.6	15.02	84.6	10.31
atom energy	3.45	18.57	21.79	-3.6E+00	-2.7E+00	107.7	7.15	77.9	38.59	74.1	7.98	71.8	7.01	60.8	12.66	60.5	8.30
lucy	61.39	24.38	31.09	-6.1E+02	1.2E+03	160.5	-	137.8	-	99.5	-	90.0	-	73.6	-	77.8	-
david1 <sub>norm</sub>	25.23	17.08	31.11	-4.4E+03	1.8E+03	322.5	-	144.9	-	155.7	-	163.4	-	108.6	-	131.9	-
torso	84.72	18.48	31.08	-2.7E+02	5.8E+02	1.9	-	1.7	-	1.5	-	1.5	-	1.3	-	1.3	-
tbl	71.90	20.14	25.99	1.5E+00	3.6E+02	8.4	-	7.1	-	5.8	-	5.6	-	4.7	-	4.8	-

Table 1. Compression results for the Miranda (m2d, m3d, M3d) structured grids, the atom point set, the lucy and david triangle meshes, and the torso and tbl tetrahedral meshes. All data but M3d is represented in single precision. The [LS2005] scheme operates on single precision only, hence the missing values. For the meshes we report only the compressed size of vertex coordinates; timings are dominated by connectivity coding, and are hence excluded. The range measures (the logarithm of) the number of floating-point values between min and max. Note that the first-order entropy is limited by the number of samples in a data set.

tion.) Arguably such data sets should use an integer rather than floating-point representation, although for simplicity or other reasons it is common practice to use floating-point. Contrary to [16], which entropy codes all bits of the residual, our new coder sacrifices such potential compression gains for speed by storing these repeated low-order bits in raw and uncompressed form. However, the massive data sets from scientific simulation that motivated our work on high-speed compression, as well as our tetrahedral meshes, rarely exhibit significant low-order redundancy, as also evidenced by our results.

### 5.1.1 Lossy Compression

Fig. 3 shows that our scheme gracefully adapts to decreasing levels of precision when discarding the least significant mantissa (and eventually exponent) bits. For  $n$  bits of precision, the schemes [7, 22] require  $\log_2 n$  bits to code the number of leading zeros, whereas our scheme exploits the combination of low entropy in the leading-zero count and the elimination of the low-order bits that are most difficult to predict and compress.

### 5.2 Compression Speed

Fig. 4 shows the speed of compressing from memory to disk, including disk write time. Because the throughput is only 20% less than our method, its decompression speed is similar to its compression speed.) We also include the raw I/O performance of dumping the data uncompressed using a single `fwrite` call. Timings

correspond to the median of five runs. Whereas our compressor is slightly slower than the less effective compressors [7, 22], it is nearly twice as fast as [16] while producing similar compression rates. However, in more I/O-intensive scenarios, such as in massively parallel simulations dumping data to the same file system (as is common), the improved compression of our method over [7, 22] results in a net gain in effective throughput. We integrated our compression code with Miranda's dump routines, and ran performance tests on 276 nodes of LLNL's Kraken supercomputer. Achieving on average a lossless reduction of 3.7 on 75 GB of data dumped, the overall dump time was reduced by a factor of 2.7 over writing the data uncompressed.

### 5.3 Entropy Coding

We compared the raw throughput of our range coder and Schindler's [23] by (1) passing raw bytes through it with no compression and (2) entropy coding byte sequences. In both cases, the source data was the uncompressed floating-point data used in our experiments. Timings show that our coder is 40% faster for raw transmission and 28% faster for entropy coding. Meanwhile, the inefficiency of our coder due to loss of precision and range reduction is only 26 bytes of overhead for 1.5 GB of coded data. Its raw throughput is only 20% less than an `fwrite` call, while its entropy coding throughput of 20 MB per second, which includes probability modeling and I/O time, compares favorably with state-of-the-art entropy coders [25].

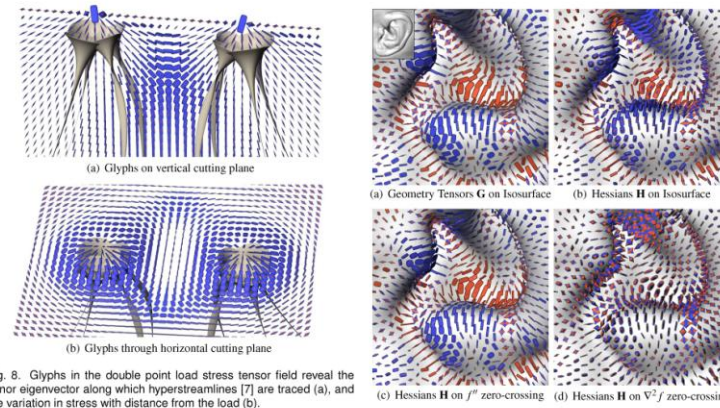


Fig. 8. Glyphs in the double point load stress tensor field reveal the minor eigenvector along which hyperstreamlines [7] are traced (a), and the variation in stress with distance from the load (b).

volume rendered [32], but its eigenvectors are commonly used in non-photo-realistic rendering, e.g. curvature-based strokes [11, 14, 19]. Inspecting geometry tensors could help debug an NPR method giving unexpected results in an unfamiliar dataset. Fig. 9(a) visualizes geometry tensors  $G$  on an isosurface (sampled by a particle system [39]) of an ear from the Visible Human male CT scan. Variations in surface curvature are reflected in the new glyphs: convex (blue circles), concave (orange circles), and saddles (orange and blue stars). For comparison, Fig. 9(b) shows the full Hessian  $H$  from which  $G$  was computed.

The new glyphs may also have a role in visualizing the tensor ingredients of image analysis methods such as edge detection. One edge definition is zero-crossing on the second directional derivative along the gradient direction,  $f'' = \mathbf{n}^T H \mathbf{n}$ . This edge surface is sampled by a particle system [33] in Fig. 9(c), showing the Hessians at the edge locations, and revealing close similarities with the geometry tensors on the isosurface in Fig. 9(a), indicating that one of the Hessian eigenvalues is near zero even though this is not part of the edge definition. Another edge definition is the zero-crossing of the Laplacian  $\nabla^2 f = \text{tr}(H)$ , and Fig. 9(d) illustrates the difference between the Hessians on this surface and those in Fig. 9(c). The consistently gray glyph halos in Fig. 9(d) indicate that these are traceless tensors.

As a demonstration of the glyphs in a 2-D visualization, Fig. 10 visualizes a cross-section of a simulation of jet flow rightward into a steady medium, causing turbulence. Glyphs of rate-of-deformations tensors document how an infinitesimal volume is stretched or compressed as it moves along the flow. A backdrop of line integral convolution [4] (with contrast modulated by velocity) provides visual context. Fig. 10(a) uses the exponentially-scaled ellipses of [34] to map tensors with negative eigenvalues to positive-definite tensors suitable for ellipsoid visualization. When the absolute difference between eigenvalues becomes too large, these glyphs can become so stretched that they overlap each other and extend over a significant portion of the domain, undermining the locality normally enjoyed by glyphs. Such stretching also reduces the visual presence of the needle-like glyphs for tensors with larger norms, contrary to scale preservation (6). Fig. 10(b) uses our superquadratic glyphs with  $\sigma(\|D\|) \approx \|D\|$ . The aspect ratio reflects the relative eigenvalue magnitudes, the size correctly indicates the tensor norm, and pointed glyphs sharply clearly communicate eigenvector directions. With compression of scale variation

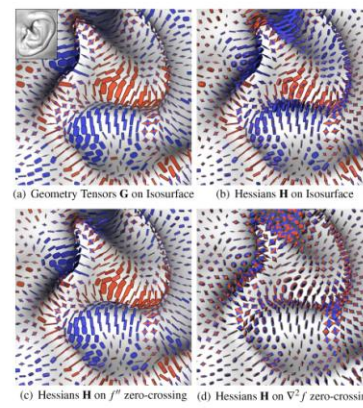


Fig. 9. Visualization of geometry tensors (a) and Hessians associated with isosurfaces (b) and two different definitions of edges, zero-crossings of the second-directional derivative (c) and the Laplacian (d). These results use  $\sigma(\|D\|) \approx \|D\|^{1/2}$  in (6).

$\sigma(\|D\|) \approx \|D\|^{1/2}$ , Fig. 10(c) better shows the directional patterns where the tensor norm is low. Colormapping the rate-of-deformation tensor trace with glyph halos highlights the regions of over-all stretching or compression, especially along the bottom edge of the domain.

Finally, Fig. 11 demonstrates how our new glyph performs traceless tensor visualization, in a side-by-side comparison to the dedicated traceless NLG tensor glyphs by Jankun-Kelly et al. [25]. Traceless tensors form a plane in eigenvalue space, and we are visualizing samples from a square within this plane, centered around the zero tensor (cf. Fig. 4(c)). Unlike the traceless glyph, which maps tensor norm to glyph sharpness, our glyph expresses norm by its overall scale  $\sigma(\|D\|) \approx \|D\|$ . Consequently, the traceless glyph requires pre-specification of maximum eigenvalue magnitudes (which are mapped to perfect sharpness), while our glyph can be used without such prior information. Another notable difference is that limiting their glyph to traceless tensors allows Jankun-Kelly et al. to make use of parts of the superquadratic shade space – including cylinders and boxes – that our approach sets aside for positive- or negative-definite tensors.

### 6 CONCLUSION

Visualization research has made significant progress in visualizing second-order tensor fields, but has mostly concentrated on the positive-definite case. Faced with indefinite tensors, a frequent strategy is to map them to positive-definite tensors prior to visualization [34, 22, 21, 52, 33]. Even when bijective mappings are used (so mathematically, no information is lost), such mappings still visually obscure the difference between positive and negative eigenvalues, which is a fundamental qualitative aspect in various applications.

Therefore, we propose an extension of a previous positive-definite tensor glyph [28] to the full space of symmetric second-order tensors. Our glyph emphasizes differences in eigenvalue sign in a way that, unlike the Reynolds glyph [18], prevents small eigenvalues from being occluded by larger ones. We also propose to use halos to ensure tensor glyph visibility even when one or more eigenvalues are near zero. Finally, we present a time- and memory-efficient implementa-

Table 1. Hypothesis tests used to reach conclusions for the dependent measures. Sharp readers will notice that one subject's TLX ratings were discarded due to not assigning any work to any of the factors, giving all techniques a workload rating of zero. Degrees of freedom for the t-tests are clamped at two less than the number of cases yielded by the study design, in which all factors were crossed for error, response time, and answers.

Test and Factor(s)	Error	Response Time	Number of Answers	Workload
ANOVA: MVV Technique	$F(3, 33) = 32.65, p = 0.00$	$F(3, 33) = 33.48, p = 0.00$	$F(3, 33) = 45.57, p = 0.00$	$F(3, 30) = 19.20, p = 0.00$
Host: DDS vs. Slivers	$t(106) = 6.59, p = 0.00$	$t(106) = 3.14, p = 0.002$	$t(106) = 0.65, p = 0.52$	$t(9) = 1.33, p = 0.22$
Host: DDS vs. Att. Blocks	$t(106) = 5.20, p = 0.00$	$t(106) = 3.23, p = 0.002$	$t(106) = 2.54, p = 0.01$	$t(9) = 1.05, p = 0.32$
ANOVA: Num. of Layers	$F(2, 22) = 7.45, p = 0.003$	$F(2, 22) = 5.37, p = 0.01$	$F(2, 22) = 1.80, p = 0.19$	$F(2, 22) = 4.30, p = 0.03$
ANOVA: Target Size	$F(2, 22) = 89.92, p = 0.00$	$F(2, 22) = 8.98, p = 0.001$	$F(2, 22) = 4.30, p = 0.03$	$F(1, 10) = 0.06, p = 0.82$
ANOVA: Experience	$F(1, 10) = 1.83, p = 0.21$	$F(1, 10) = 6.17, p = 0.03$	$F(1, 10) = 0.06, p = 0.82$	$F(4, 44) = 9.06, p = 0.00$
ANOVA: MVV-by-Num. Layers	$F(4, 44) = 8.79, p = 0.00$	$F(4, 44) = 35.64, p = 0.00$	$F(4, 44) = 3.15, p = 0.02$	-

Table 2. The mean and standard deviation for each MVV technique for each of the three objective dependent measures and the subjective workload rating shows the difficulty users had in attempting to complete the task with the baseline technique. Error is expressed in units of layers (range: 0-6), time in seconds, answers in a count, and workload through NASA TLX.

Name	Error (layers)	Error Std. Dev.	Time (sec)	Time Std. Dev.	Answers (count)	Answers Std. Dev.	Workload (TLX)	Workload Std. Dev.
JuxLayers	1.54	1.03	46.72	36.44	8.87	8.09	65.53	15.40
DDSpots	0.09	0.21	7.33	6.01	1.09	0.28	40.74	23.17
Slivers	0.42	0.48	9.62	4.25	1.11	0.24	27.86	22.38
Attrib	0.47	0.73	9.74	5.27	1.20	0.37	31.79	16.24

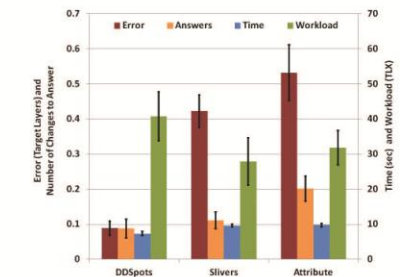


Fig. 6. Graph of dependent measures for the integrated MVV techniques. There was a main effect on error (red) and workload (green), but not on time (blue) or answers selected (orange). Error and number of changes to answers (i.e. one less than the number of answers selected, to align the graphs better) are on the primary axis on the left. Response time and workload are on the secondary axis on the right.

that stated only that six layers would be the most difficult and not predicted a complete ordering with respect to increasing number of layers. Further, we found that users were 13% faster with six layers in the target than with five layers, which is a bit counter-intuitive and a result that we shall discuss in Section 4.

### 3.5.4 Effect of Size of Target

The size of the target had a main effect on error, response time, and the number of answers selected. However, the results again do not support the complete ordering predicted in Hypothesis 4. The smallest target size was clearly more difficult, but there was no significant difference between the two larger sizes. Users were fastest with the largest target size, with a small but not significant difference between the smallest and middle sizes. Users changed their answers at a slightly increasing rate with decreasing target size.

### 3.5.5 Effect of User Experience

We expected (based on our past work) to see users who had participated in previous studies perform faster. We found a main effect of (binary) user experience on response time (Table 1). Returning users were on average 30.7% faster than subjects who were participating in our sequence of studies for the first time. This confirms Hypothesis 5.

### 3.5.6 Other Findings

We found significant interactions between MVV Technique and the number of target layers for error and for response time. We found significant interactions between MVV Technique and the target size for error and for response time. Since these results (Table 1) give us insight into the usability of the techniques and also implicitly show the main effects of the number of target layers and of the target size, we graph these results in Figure 7.

There was a significant interaction between the number of target layers and the target size for  $F(4, 44) = 5.128, p = 0.002$ . For all number of target layers, the smallest targets were most difficult, but the magnitude of the increase in difficulty from the middle size down to the smallest size was quite a bit lower for five layers than would be expected looking at the jumps for four and six layers.

We checked whether fatigue had an effect on error by running a 3 (MVV Technique)  $\times$  36 (Count) ANOVA with the MVV technique and the count of questions as factors; we found no significant effect of the count of questions completed –  $F(35, 385) = 0.798, p = 0.789$ . Similarly, we conducted a 3 (MVV Technique)  $\times$  3 (Target Layers)  $\times$  3 (Target Size)  $\times$  4 (Repetition) ANOVA to see if repetition of the combination of target size and number of target layers had a main effect; we found no significant effect –  $F(3, 33) = 0.860, p = 0.472$ . Analogous ANOVA calculations revealed that there was no significant effect of trial count or repetition on the number of answers selected.

We ran a filter on the error to find trials where the response was judged to be incorrect, but the error in pixels from the correct answer was smaller than the size of the target. There were only nine such errors in 494 trials that saw errors (out of 1728 total trials), so we cannot attach statistical significance to the occurrence of such an event. But we do find it curious to note that of the nine such errors, seven saw selections that were no more than seven pixels away from the target patch – and all of these were trials with Attribute Blocks and the smallest target size (31 pixels). (Two trials saw selections that were almost the size of the target patch – 61 or 91 pixels, respectively – with Oriented Slivers.)



# Results: evaluation scenarios (selection)

algorithmic performance  
35% of scenarios

qualitative result inspection  
46% of scenarios

user performance/experience  
14% of scenarios

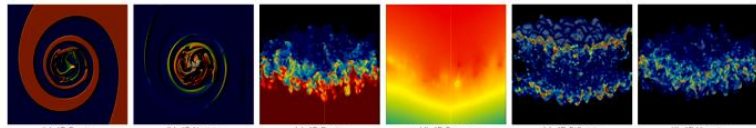


Fig. 1. Visualizations of 2D data (as pseudocolored height fields) and 3D data (volume rendered) used in our experiments.

name	data set		compressed size (MB) and compression time (seconds)														
	unique (%)	entropy (bits)	min	max	size (MB)	time (sec)	zlib	[RKB2006]	[EFF2000]	[ILS2005]	new scheme						
m2d density	3.89	3.49	21.83	8.7E-01	1.2E+00	19.6	0.71	1.6	0.86	4.3	0.49	4.4	0.56	1.3	1.08	1.3	0.56
m2d vorticity	99.20	22.25	31.05	-1.4E+02	2.5E+01	19.6	0.71	18.4	2.14	11.8	1.21	15.5	1.29	12.9	2.22	13.8	1.49
m3d density	7.67	5.16	23.60	1.0E+00	3.0E+00	364.5	12.81	50.4	17.55	100.5	9.06	96.3	8.48	35.7	19.03	35.5	9.25
m3d pressure	27.29	23.91	31.06	-3.7E+00	2.3E+03	364.5	12.80	229.2	99.76	95.6	9.31	87.9	8.87	40.1	18.79	40.4	9.96
m3d diffusivity	36.87	23.19	30.02	0.0E+00	6.8E+00	364.5	12.68	297.6	42.90	250.8	19.08	239.3	15.02	198.8	31.92	203.0	18.47
m3d viscosity	50.07	24.86	28.59	8.6E-15	2.9E+00	364.5	12.62	314.0	46.09	249.4	18.95	246.1	14.68	209.2	32.66	207.5	19.45
h3d temp	65.70	23.54	31.56	-7.7E+01	1.0E+35	95.4	3.77	75.8	14.56	59.3	4.64	53.0	4.27	44.1	8.04	44.1	5.06
h3d pressure	81.82	24.13	31.58	-3.4E+03	1.0E+35	95.4	3.78	82.3	12.00	64.3	5.14	52.9	4.87	45.0	7.78	45.2	5.34
h3d x velocity	84.18	24.18	31.55	-5.3E+01	1.0E+35	95.4	3.89	86.1	11.27	67.4	6.22	63.3	4.59	54.5	8.86	55.4	5.44
h3d y velocity	84.32	24.18	31.55	-4.6E+01	1.0E+35	95.4	3.83	84.5	11.42	67.1	5.74	62.3	5.04	53.5	8.64	53.8	5.53
h3d z velocity	86.82	24.24	31.54	-3.2E+00	1.0E+35	95.4	3.87	88.4	10.76	85.6	8.50	76.9	5.29	68.9	9.83	69.1	6.65
M3d density	40.14	18.81	22.59	1.0E+00	3.0E+00	288.0	11.28	136.8	41.91	160.3	11.63	121.6	10.94	-	-	105.2	11.63
M3d pressure	100.00	25.17	63.00	-2.2E+00	2.2E+00	288.0	11.20	272.6	35.18	237.3	14.91	225.1	16.59	-	-	208.4	17.20
M3d x velocity	100.00	25.17	63.00	-2.2E+00	2.3E+00	288.0	10.83	275.6	32.30	230.4	14.73	215.1	15.91	-	-	197.7	16.84
M3d y velocity	100.00	25.17	63.00	-2.1E+00	2.3E+00	288.0	10.54	275.1	32.19	223.1	14.27	215.2	15.16	-	-	197.7	16.65
M3d z velocity	100.00	25.17	63.00	-5.2E+00	9.0E+00	288.0	10.32	275.5	32.62	226.6	14.74	213.7	16.05	-	-	196.8	16.14
atom x position	61.10	23.82	31.01	-4.8E-02	4.6E+02	107.7	7.07	84.3	21.18	76.0	7.88	78.8	7.61	67.3	12.88	68.6	9.07
atom y position	45.90	23.32	26.99	3.7E-02	2.1E+03	107.7	7.08	65.9	30.76	60.4	6.97	56.4	6.31	47.0	10.49	46.9	7.73
atom z position	61.68	23.84	27.48	9.1E-05	4.6E+02	107.7	7.46	94.6	19.86	82.6	9.00	86.1	8.25	75.7	13.80	78.2	9.93
atom x velocity	64.65	23.87	30.96	1.5E-01	1.4E-01	107.7	7.30	95.7	19.88	93.8	10.07	99.1	9.65	84.8	14.93	87.6	9.92
atom temp	64.91	23.94	27.41	3.0E-03	7.1E+03	107.7	6.69	95.7	19.76	91.6	10.27	95.9	8.34	84.6	15.02	84.6	10.31
atom energy	3.45	18.57	21.79	-3.6E+00	-2.7E+00	107.7	7.15	77.9	38.59	74.1	7.98	71.8	7.01	60.8	12.66	60.5	8.30
lucy	61.39	24.38	31.09	-6.1E+02	1.2E+03	160.5	-	137.8	-	99.5	-	90.0	-	73.6	-	77.8	-
david10m	25.23	17.08	31.11	-4.4E+03	1.8E+03	322.5	-	144.9	-	155.7	-	163.4	-	108.6	-	131.9	-
torso	84.72	18.48	31.08	-2.7E+02	5.8E+02	1.9	-	1.7	-	1.5	-	1.5	-	1.3	-	1.3	-
rhl	71.90	20.14	25.99	1.5E+00	3.6E+02	8.4	-	7.1	-	5.8	-	5.6	-	4.7	-	4.8	-

Table 1. Compression results for the Miranda (m2d, m3d, M3d) structured grids, the atom point set, the lucy and david triangle meshes, and the torso and rhl tetrahedral meshes. All data but M3d is represented in single precision. The [ILS2005] scheme operates on single precision only, hence the missing values. For the meshes we report only the compressed size of vertex coordinates; timings are dominated by connectivity coding, and are hence excluded. The range measures (the logarithm of) the number of floating-point values between min and max. Note that the first-order entropy is limited by the number of samples in a data set.

ation.) Arguably such data sets should use an integer rather than floating-point representation, although for simplicity or other reasons it is common practice to use floating-point. Contrary to [16], which entropy codes all bits of the residual, our new coder sacrifices such potential compression gains for speed by storing these repeated low-order bits in raw and uncompressed form. However, the massive data sets from scientific simulation that motivated our work on high-speed compression, as well as our tetrahedral meshes, rarely exhibit such significant low-order redundancy, as also evidenced by our results.

### 5.1.1 Lossy Compression

Fig. 3 shows that our scheme gracefully adapts to decreasing levels of precision when discarding the least significant mantissa (and eventually exponent) bits. For  $n$  bits of precision, the schemes [7, 22] require  $\log_2 n$  bits to code the number of leading zeros, whereas our scheme exploits the combination of low entropy in the leading-zero count and the elimination of the low-order bits that are most difficult to predict and compress.

### 5.2 Compression Speed

Fig. 4 shows the speed of compressing from memory to disk, including disk write time. Because the throughput is only 20% less than our method, its decompression speed is similar to its compression speed.) We also include the raw I/O performance of dumping the data uncompressed using a single `fwrite` call. Timings

correspond to the median of five runs. Whereas our compressor is slightly slower than the less effective compressors [7, 22], it is nearly twice as fast as [16] while producing similar compression rates. However, in more I/O-intensive scenarios, such as in massively parallel simulations dumping data to the same file system (as is common), the improved compression of our method over [7, 22] results in a net gain in effective throughput. We integrated our compression code with Miranda's dump routines, and ran performance tests on 276 nodes of LLNL's ACER supercomputer. Achieving on average a lossless reduction of 3.7 on 75 GB of data dumped, the overall dump time was reduced by a factor of 2.7 over writing the data uncompressed.

### 5.3 Entropy Coding

We compared the raw throughput of our range coder and Schindler's [23] by (1) passing raw bytes through it with no compression and (2) entropy coding byte sequences. In both cases, the source data was the uncompressed floating-point data used in our experiments. Timings show that our coder is 40% faster for raw transmission and 28% faster for entropy coding. Meanwhile, the inefficiency of our coder due to loss of precision and range reduction is only 26 bytes of overhead for 1.5 GB of coded data. Its raw throughput is only 20% less than an `fwrite` call, while its entropy coding throughput of 20 MB per second, which includes probability modeling and I/O time, compares favorably with state-of-the-art entropy coders [25].

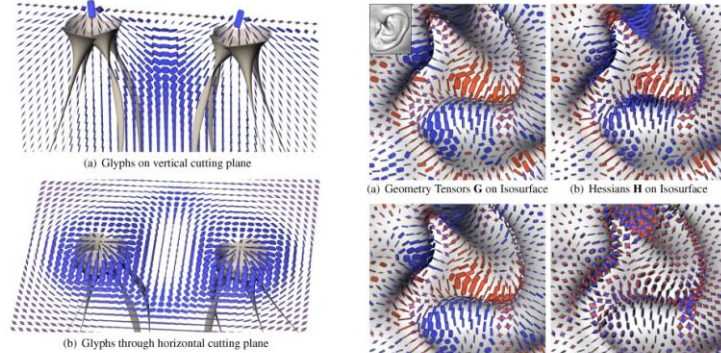


Fig. 8. Glyphs in the double point load stress tensor field reveal the minor eigenvector along which hyperstreamlines [7] are traced (a), and the variation in stress with distance from the load (b).

volume rendered [32], but its eigenvectors are commonly used in non-photo-realistic rendering, e.g. curvature-based strokes [11, 14, 19]. Inspecting geometry tensors could help debug an NPR method giving unexpected results in an unfamiliar dataset. Fig. 9(a) visualizes geometry tensors  $G$  on an isosurface (sampled by a particle system [39]) of an ear from the Visible Human male CT scan. Variations in surface curvature are reflected in the new glyphs: convex (blue circles), concave (orange circles), and saddles (orange and blue stars). For comparison, Fig. 9(b) shows the full Hessian  $H$  from which  $G$  was computed.

The new glyphs may also have a role in visualizing the tensor ingredients of image analysis methods such as edge detection. One edge definition is zero-crossing on the second directional derivative along the gradient direction,  $f'' = \mathbf{n}^T \mathbf{H} \mathbf{n}$ . This edge surface is sampled by a particle system [33] in Fig. 9(c), showing the Hessians at the edge locations, and revealing close similarities with the geometry tensors on the isosurface in Fig. 9(a), indicating that one of the Hessian eigenvalues is near zero even though this is not part of the edge definition. Another edge definition is the zero-crossing of the Laplacian  $\nabla^2 f = \text{tr}(\mathbf{H})$ , and Fig. 9(d) illustrates the difference between the Hessians on this surface and those in Fig. 9(c). The consistently gray glyph halos in Fig. 9(d) indicate that these are traceless tensors.

As a demonstration of the glyphs in a 2-D visualization, Fig. 10 visualizes a cross-section of a simulation of jet flow rightward into a steady medium, causing turbulence. Glyphs of rate-of-deformations tensors document how an infinitesimal volume is stretched or compressed as it moves along the flow. A backdrop of line integral convolution [4] (with contrast modulated by velocity) provides visual context. Fig. 10(a) uses the exponentially-scaled ellipses of [34] to map tensors with negative eigenvalues to positive-definite tensors suitable for ellipsoid visualization. When the absolute difference between eigenvalues becomes too large, these glyphs can become so stretched that they overlap each other and extend over a significant portion of the domain, undermining the locality normally enjoyed by glyphs. Such stretching also reduces the visual presence of the needle-like glyphs for tensors with larger norms, contrary to scale preservation (6). Fig. 10(b) uses our superquadratic glyphs with  $\sigma(\|\mathbf{D}\|) \approx \|\mathbf{D}\|$ . The aspect ratio reflects the relative eigenvalue magnitudes, the size correctly indicates the tensor norm, and pointed glyphs sharply clearly communicate eigenvector directions. With compression of scale variation

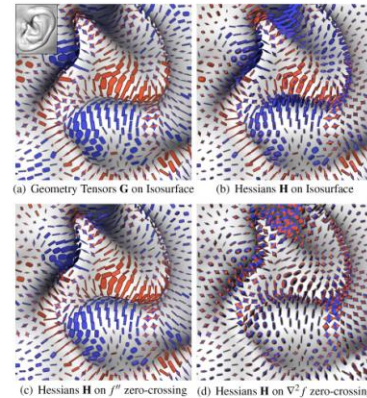


Fig. 9. Visualization of geometry tensors (a) and Hessians associated with isosurfaces (b) and two different definitions of edges, zero-crossings of the second directional derivative (c) and the Laplacian (d). These results use  $\sigma(\|\mathbf{D}\|) \approx \|\mathbf{D}\|^{1/2}$  in (6).

$\sigma(\|\mathbf{D}\|) \approx \|\mathbf{D}\|^{1/2}$ , Fig. 10(c) better shows the directional patterns where the tensor norm is low. Colormapping the rate-of-deformation tensor trace with glyph halos highlights the regions of over-all stretching or compression, especially along the bottom edge of the domain.

Finally, Fig. 11 demonstrates how our new glyph performs traceless tensor visualization, in a side-by-side comparison to the dedicated traceless NLG tensor glyphs by Jankun-Kelly et al. [25]. Traceless tensors form a plane in eigenvalue space, and we are visualizing samples from a square within this plane, centered around the zero tensor (cf. Fig. 4(c)). Unlike the traceless glyph, which maps tensor norm to glyph sharpness, our glyph expresses norm by its overall scale  $\sigma(\|\mathbf{D}\|) \approx \|\mathbf{D}\|$ . Consequently, the traceless glyph requires pre-specification of maximum eigenvalue magnitudes (which are mapped to perfect sharpness), while our glyph can be used without such prior information. Another notable difference is that limiting their glyph to traceless tensors allows Jankun-Kelly et al. to make use of parts of the superquadratic shape space - including cylinders and boxes - that our approach sets aside for positive- or negative-definite tensors.

### 6 CONCLUSION

Visualization research has made significant progress in visualizing second-order tensor fields, but has mostly concentrated on the positive-definite case. Faced with indefinite tensors, a frequent strategy is to map them to positive-definite tensors prior to visualization [34, 22, 21, 52, 33]. Even when bijective mappings are used (so mathematically, no information is lost), such mappings still visually obscure the difference between positive and negative eigenvalues, which is a fundamental qualitative aspect in various applications.

Therefore, we propose an extension of a previous positive-definite tensor glyph [28] to the full space of symmetric second-order tensors. Our glyph emphasizes differences in eigenvalue sign in a way that, unlike the Reynolds glyph [18], prevents small eigenvalues from being occluded by larger ones. We also propose to use halos to ensure tensor glyph visibility even when one or more eigenvalues are near zero. Finally, we present a time- and memory-efficient implementa-

Table 1. Hypothesis tests used to reach conclusions for the dependent measures. Sharp results will notice that one subject's TLX ratings were discarded due to not assigning any work to any of the factors, giving all techniques a workload rating of zero. Degrees of freedom for the t-tests are clamped at two less than the number of cases yielded by the study design, in which all factors were crossed for error, response time, and answers.

Test and Factor(s)	Error	Response Time	Number of Answers	Workload
ANOVA: MVV Technique	$F(3, 33) = 32.65, p = 0.00$	$F(3, 33) = 35.48, p = 0.00$	$F(3, 33) = 45.57, p = 0.00$	$F(3, 30) = 19.20, p = 0.00$
Post: DDS vs. Slivers	$t(106) = 6.59, p = 0.00$	$t(106) = 3.14, p = 0.002$	$t(106) = 0.65, p = 0.52$	$t(9) = 1.33, p = 0.22$
Post: DDS vs. Att. Blocks	$t(106) = 5.20, p = 0.00$	$t(106) = 3.23, p = 0.002$	$t(106) = 2.54, p = 0.01$	$t(9) = 1.05, p = 0.32$
ANOVA: Num. of Layers	$F(2, 22) = 7.45, p = 0.003$	$F(2, 22) = 5.37, p = 0.01$	$F(2, 22) = 1.80, p = 0.19$	$F(2, 22) = 4.30, p = 0.03$
ANOVA: Target Size	$F(2, 22) = 89.92, p = 0.00$	$F(2, 22) = 8.98, p = 0.001$	$F(2, 22) = 6.17, p = 0.03$	$F(1, 10) = 0.06, p = 0.82$
ANOVA: Experience	$F(1, 10) = 1.83, p = 0.21$	$F(1, 10) = 6.17, p = 0.03$	$F(1, 10) = 0.06, p = 0.82$	$F(1, 10) = 0.06, p = 0.82$
ANOVA: MVV-by-Num. Layers	$F(4, 44) = 8.79, p = 0.00$	$F(4, 44) = 9.06, p = 0.00$	$F(4, 44) = 3.15, p = 0.02$	$F(4, 44) = 3.15, p = 0.02$

Table 2. The mean and standard deviation for each MVV technique for each of the three objective dependent measures and the subjective workload rating shows the difficulty users had in attempting to complete the task with the baseline technique. Error is expressed in units of layers (range 0-6), time in seconds, answers in a count, and workload through NASA TLX.

Name	Error (layers)	Std. Dev.	Time (sec)	Time (sec)	Answers (count)	Answers (Std. Dev.)	Workload (TLX)	Std. Dev.
JuxLayers	1.54	1.03	46.72	36.44	8.87	8.09	65.53	15.40
DDSpots	0.09	0.21	7.33	6.01	1.09	0.28	40.74	23.17
Slivers	0.42	0.48	9.62	4.25	1.11	0.24	27.86	22.38
Attrib	0.47	0.73	9.74	5.27	1.20	0.37	31.79	16.24

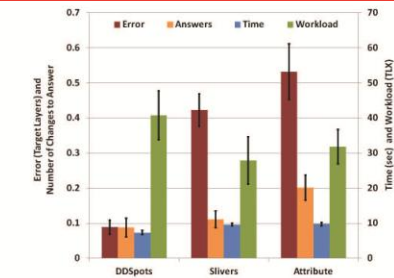


Fig. 6. Graph of dependent measures for the integrated MVV techniques. There was a main effect on error (red) and workload (green), but not on time (blue) or answers selected (orange). Error and number of changes to answers (i.e. one less than the number of answers selected, to align the graphs better) are on the primary axis on the left. Response time and workload are on the secondary axis on the right.

that stated only that six layers would be the most difficult and not predicted a complete ordering with respect to increasing number of layers. Further, we found that users were 13% faster with six layers in the target than with five layers, which is a bit counter-intuitive and a result that we shall discuss in Section 4.

### 3.5.4 Effect of Size of Target

The size of the target had a main effect on error, response time, and the number of answers selected. However, the results again do not support the complete ordering predicted in Hypothesis 4. The smallest target size was clearly more difficult, but there was no significant difference between the two larger sizes. Users were fastest with the largest target size, with a small but not significant difference between the smallest and middle sizes. Users changed their answers at a slightly increasing rate with decreasing target size.

### 3.5.5 Effect of User Experience

We expected (based on our past work) to see users who had participated in previous studies perform faster. We found a main effect of (binary) user experience on response time (Table 1). Returning users were on average 30.7% faster than subjects who were participating in our sequence of studies for the first time. This confirms Hypothesis 5.

### 3.5.6 Other Findings

We found significant interactions between MVV Technique and the number of target layers for error and for response time. We found significant interactions between MVV Technique and the target size for error and for response time. Since these results (Table 1) give us insight into the usability of the techniques and also implicitly show the main effects of the number of target layers and of the target size, we graph these results in Figure 7.

There was a significant interaction between the number of target layers and the target size for  $-F(4, 44) = 5.128, p = 0.002$ . For all number of target layers, the smallest targets were most difficult, but the magnitude of the increase in difficulty from the middle size down to the smallest size was quite a bit lower for five layers than would be expected looking at the jumps for four and six layers.

We checked whether fatigue had an effect on error by running a 3 (MVV Technique)  $\times$  36 (Count) ANOVA with the MVV technique and the count of questions as factors; we found no significant effect of the count of questions completed -  $F(35, 385) = 0.798, p = 0.789$ . Similarly, we conducted a 3 (MVV Technique)  $\times$  3 (Target Layers)  $\times$  3 (Target Size)  $\times$  4 (Repetition) ANOVA to see if repetition of the combination of target size and number of target layers had a main effect; we found no significant effect -  $F(3, 33) = 0.860, p = 0.472$ . Analogous ANOVA calculations revealed that there was no significant effect of trial count or repetition on the number of answers selected.

We ran a filter on the error to find trials where the response was judged to be incorrect, but the error in pixels from the correct answer was smaller than the size of the target. There were only nine such errors in 494 trials that saw errors (out of 1728 total trials), so we cannot attach statistical significance to the occurrence of such an event. But we do find it curious to note that of the nine such errors, seven saw selections that were no more than seven pixels away from the target patch - and all of these were trials with Attribute Blocks and the smallest target size (31 pixels). (Two trials saw selections that were almost the size of the target patch - 61 or 91 pixels, respectively - with Oriented Slivers.)

# Results: evaluation scenarios (selection)

algorithmic performance  
35% of scenarios

qualitative result inspection  
46% of scenarios

user performance/experience  
14% of scenarios

LINDSTROM et al.: FAST AND EFFICIENT COMPRESSION OF FLOATING-POINT DATA 1249

Fig. 1. Visualizations of 2D data (as pseudocolored height fields) and 3D data (volume rendered) used in our experiments.

name	unique entropy range		compressed size (MB)		compression time (seconds)		compressed size (MB) and compression time (seconds)										
	(%)	(bits)	min	max	size (MB)	time (sec)	zlib	[RKB2006]	[EFF2000]	[ILS2005]	new scheme						
m2d density	3.89	3.49	21.83	8.7E-01	1.2E+00	19.6	0.71	1.6	0.86	4.3	0.49	4.4	0.56	1.3	1.08	1.3	0.56
m2d vorticity	99.20	22.25	31.05	-1.4E+02	2.5E+01	19.6	0.71	18.4	2.14	11.8	1.21	15.5	1.29	12.9	2.22	13.8	1.49
m3d density	7.67	5.16	23.60	1.0E+00	3.0E+00	364.5	12.81	50.4	17.55	100.5	9.06	96.3	8.48	35.7	19.03	35.5	9.25
m3d pressure	27.29	23.91	31.06	-3.7E-00	2.3E+03	364.5	12.80	229.2	99.76	95.6	9.31	87.9	8.87	40.1	18.79	40.4	9.96
m3d diffusivity	36.87	23.19	30.02	0.0E+00	6.8E+00	364.5	12.68	297.6	42.90	250.8	19.09	239.3	15.02	198.8	31.92	203.0	18.47
m3d viscosity	50.07	24.86	28.59	8.6E-15	2.9E+00	364.5	12.62	314.0	46.09	249.4	18.95	246.1	14.68	209.2	32.66	207.5	19.45
h3d temp	65.70	23.54	31.56	-7.7E+01	1.0E+35	95.4	3.77	75.8	14.56	59.3	4.64	53.0	4.27	44.1	8.04	44.1	5.06
h3d pressure	81.82	24.13	31.58	-3.4E+03	1.0E+35	95.4	3.78	82.3	12.00	64.3	5.14	52.9	4.87	45.0	7.78	45.2	5.34
h3d x velocity	84.18	24.18	31.55	-5.3E+01	1.0E+35	95.4	3.89	86.1	11.27	67.4	6.22	63.3	4.59	54.5	8.86	55.4	5.44
h3d y velocity	84.32	24.18	31.55	-4.6E+01	1.0E+35	95.4	3.83	84.5	11.42	67.1	5.74	62.3	5.04	53.5	8.64	53.8	5.53
h3d z velocity	86.82	24.24	31.54	-3.2E+00	1.0E+35	95.4	3.87	88.4	10.76	85.6	8.50	76.9	5.29	68.9	9.83	69.1	6.65
M3d density	40.14	18.81	22.59	1.0E+00	3.0E+00	288.0	11.28	136.8	41.91	160.3	11.63	121.6	10.94	-	-	105.2	11.63
M3d pressure	100.00	25.17	63.00	-2.2E+00	2.2E+00	288.0	11.20	272.6	35.18	237.3	14.91	225.1	16.59	-	-	208.4	17.20
M3d x velocity	100.00	25.17	63.00	-2.2E+00	2.3E+00	288.0	10.83	275.6	32.30	230.4	14.73	215.1	15.91	-	-	197.7	16.84
M3d y velocity	100.00	25.17	63.00	-2.1E+00	2.3E+00	288.0	10.54	275.1	32.19	223.1	14.27	215.2	15.16	-	-	197.7	16.65
M3d z velocity	100.00	25.17	63.00	-5.2E+00	9.0E+00	288.0	10.32	275.5	32.62	226.6	14.74	213.7	16.05	-	-	196.8	16.14
atom x position	61.10	23.82	31.01	-4.8E-02	4.6E+02	107.7	7.07	84.3	21.18	76.0	7.88	78.8	7.61	67.3	12.88	68.6	9.07
atom y position	45.90	23.32	26.99	3.7E-02	2.1E+03	107.7	7.08	65.9	30.76	60.4	6.97	56.4	6.31	47.0	10.49	46.9	7.73
atom z position	61.68	23.84	27.48	9.1E-05	4.6E+02	107.7	7.46	94.6	19.86	82.6	9.00	86.1	8.25	75.7	13.80	78.2	9.93
atom x velocity	64.65	23.87	30.96	1.5E-01	1.4E-01	107.7	7.30	95.7	19.88	93.8	10.07	99.1	9.65	84.8	14.93	87.6	9.92
atom temp	64.91	23.94	27.41	3.0E-03	7.1E+03	107.7	6.69	95.7	19.76	91.6	10.27	95.9	8.34	84.6	15.02	84.6	10.31
atom energy	3.45	18.57	21.79	-3.6E+00	-2.7E+00	107.7	7.15	77.9	38.59	74.1	7.98	71.8	7.01	60.8	12.66	60.5	8.30
lucy	61.39	24.38	31.09	-6.1E+02	1.2E+03	160.5	-	137.8	-	99.5	-	90.0	-	73.6	-	77.8	-
david1 <sub>norm</sub>	25.23	17.08	31.11	-4.4E+03	1.8E+03	322.5	-	144.9	-	155.7	-	163.4	-	108.6	-	131.9	-
torso	84.72	18.48	31.08	-2.7E+02	5.8E+02	1.9	-	1.7	-	1.5	-	1.5	-	1.3	-	1.3	-
tbl	71.90	20.14	25.99	1.5E+00	3.6E+02	8.4	-	7.1	-	5.8	-	5.6	-	4.7	-	4.8	-

Table 1. Compression results for the Miranda (m2d, m3d, M3d) structured grids, the atom point set, the lucy and david triangle meshes, and the torso and tbl tetrahedral meshes. All data but M3d is represented in single precision. The [ILS2005] scheme operates on single precision only, hence the missing values. For the meshes we report only the compressed size of vertex coordinates; timings are dominated by connectivity coding, and are hence excluded. The range measures (the logarithm of) the number of floating-point values between min and max. Note that the first-order entropy is limited by the number of samples in a data set.

tion.) Arguably such data sets should use an integer rather than floating-point representation, although for simplicity or other reasons it is common practice to use floating-point. Contrary to [16], which entropy codes all bits of the residual, our new coder sacrifices such potential compression gains for speed by storing these repeated low-order bits in raw and uncompressed form. However, the massive data sets from scientific simulation that motivated our work on high-speed compression, as well as our tetrahedral meshes, rarely exhibit significant low-order redundancy, as also evidenced by our results.

5.1.1 Lossy Compression

Fig. 3 shows that our scheme gracefully adapts to decreasing levels of precision when discarding the least significant mantissa (and eventually exponent) bits. For  $n$  bits of precision, the schemes [7, 22] require  $\log_2 n$  bits to code the number of leading zeros, whereas our scheme exploits the combination of low entropy in the leading-zero count and the elimination of the low-order bits that are most difficult to predict and compress.

5.2 Compression Speed

Fig. 4 shows the speed of compressing from memory to disk, including disk write time. Because the simplicity of our method, its decompression speed is similar to its compression speed.) We also include the raw I/O performance of dumping the data uncompressed using a single `fwrite` call. Timings

correspond to the median of five runs. Whereas our compressor is slightly slower than the less effective compressors [7, 22], it is nearly twice as fast as [16] while producing similar compression rates. However, in more I/O-intensive scenarios, such as in massively parallel simulations dumping data to the same file system (as is common), the improved compression of our method over [7, 22] results in a net gain in effective throughput. We integrated our compression code with Miranda's dump routines, and ran performance tests on 276 nodes of LLNL's ACES supercomputer. Achieving on average a lossless reduction of 3.7 on 75 GB of data dumped, the overall dump time was reduced by a factor of 2.7 over writing the data uncompressed.

5.3 Entropy Coding

We compared the raw throughput of our range coder and Schindler's [23] by (1) passing raw bytes through it with no compression and (2) entropy coding byte sequences. In both cases, the source data is the uncompressed floating-point data used in our experiments. Timings show that our coder is 40% faster for raw transmission and 28% faster for entropy coding. Meanwhile, the inefficiency of our coder due to loss of precision and range reduction is only 26 bytes of overhead for 1.5 GB of coded data. Its raw throughput is only 20% less than an `fwrite` call, while its entropy coding throughput of 20 MB per second, which includes probability modeling and I/O time, compares favorably with state-of-the-art entropy coders [25].

1602 IEEE TRANSACTIONS ON VISUALIZATION AND COMPUTER GRAPHICS, VOL. 16, NO. 6, NOVEMBER/DECEMBER 2010

Fig. 8. Glyphs in the double point load stress tensor field reveal the minor eigenvector along which hyperstreamlines [7] are traced (a), and the variation in stress with distance from the load (b).

volume rendered [32], but its eigenvectors are commonly used in non-photo-realistic rendering, e.g. curvature-based strokes [11, 14, 19]. Inspecting geometry tensors could help debug an NPR method giving unexpected results in an unfamiliar dataset. Fig. 9(a) visualizes geometry tensors  $G$  on an isosurface (sampled by a particle system [39]) of an ear from the Visible Human male CT scan. Variations in surface curvature are reflected in the new glyphs: convex (blue circles), concave (orange circles), and saddles (orange and blue stars). For comparison, Fig. 9(b) shows the full Hessian  $H$  from which  $G$  was computed.

The new glyphs may also have a role in visualizing the tensor ingredients of image analysis methods such as edge detection. One edge definition is zero-crossing on the second directional derivative along the gradient direction,  $f'' = \mathbf{n}^T H \mathbf{n}$ . This edge surface is sampled by a particle system [33] in Fig. 9(c), showing the Hessians at the edge locations, and revealing close similarities with the geometry tensors on the isosurface in Fig. 9(a), indicating that one of the Hessian eigenvalues is near zero even though this is not part of the edge definition. Another edge definition is the zero-crossing of the Laplacian  $\nabla^2 f = \text{tr}(H)$ , and Fig. 9(d) illustrates the difference between the Hessians on this surface and those in Fig. 9(c). The consistently gray glyph halos in Fig. 9(d) indicate that these are traceless tensors.

As a demonstration of the glyphs in a 2-D visualization, Fig. 10 visualizes a cross-section of a simulation of jet flow rightward into a steady medium, causing turbulence. Glyphs of rate-of-deformations tensors document how an infinitesimal volume is stretched or compressed as it moves along the flow. A backdrop of line integral convolution [4] (with contrast modulated by velocity) provides visual context. Fig. 10(a) uses the exponentially-scaled ellipses of [34] to map tensors with negative eigenvalues to positive-definite tensors suitable for ellipsoid visualization. When the absolute difference between eigenvalues becomes too large, these glyphs can become so stretched that they overlap each other and extend over a significant portion of the domain, undermining the locality normally enjoyed by glyphs. Such stretching also reduces the visual presence of the needle-like glyphs for tensors with larger norms, contrary to scale preservation (6). Fig. 10(b) uses our superquadratic glyphs with  $s(\|D\|) \propto \|D\|$ . The aspect ratio reflects the relative eigenvalue magnitudes, the size correctly indicates the tensor norm, and pointed glyphs sharply clearly communicate eigenvector directions. With compression of scale variation

Fig. 9. Visualization of geometry tensors (a) and Hessians associated with isosurfaces (b) and two different definitions of edges, zero-crossings of the second directional derivative (c) and the Laplacian (d). These results use  $s(\|D\|) \propto \|D\|^{1/2}$  in (6).

(a) Geometry Tensors  $G$  on Isosurface (b) Hessians  $H$  on Isosurface

(c) Hessians  $H$  on  $f''$  zero-crossing (d) Hessians  $H$  on  $\nabla^2 f$  zero-crossing

Fig. 9. Visualization of geometry tensors (a) and Hessians associated with isosurfaces (b) and two different definitions of edges, zero-crossings of the second directional derivative (c) and the Laplacian (d). These results use  $s(\|D\|) \propto \|D\|^{1/2}$  in (6).

(a) Geometry Tensors  $G$  on Isosurface (b) Hessians  $H$  on Isosurface

(c) Hessians  $H$  on  $f''$  zero-crossing (d) Hessians  $H$  on  $\nabla^2 f$  zero-crossing

Fig. 11 demonstrates how our new glyph performs traceless tensor visualization, in a side-by-side comparison to the dedicated traceless NLG tensor glyphs by Jankun-Kelly et al. [25]. Traceless tensors form a plane in eigenvalue space, and we are visualizing samples from a square within this plane, centered around the zero tensor (cf. Fig. 4(c)). Unlike the traceless glyph, which maps tensor norm to glyph sharpness, our glyph expresses norm by its overall scale  $s(\|D\|) \propto \|D\|$ . Consequently, the traceless glyph requires pre-specification of maximum eigenvalue magnitudes (which are mapped to perfect sharpness), while our glyph can be used without such prior information. Another notable difference is that limiting their glyph to traceless tensors allows Jankun-Kelly et al. to make use of parts of the superquadratic shape space – including cylinders and boxes – that our approach sets aside for positive- or negative-definite tensors.

6 CONCLUSION

Visualization research has made significant progress in visualizing second-order tensor fields, but has mostly concentrated on the positive-definite case. Faced with indefinite tensors, a frequent strategy is to map them to positive-definite tensors prior to visualization [34, 22, 21, 52, 33]. Even when bijective mappings are used (so mathematically, no information is lost), such mappings still visually obscure the difference between positive and negative eigenvalues, which is a fundamental qualitative aspect in various applications.

Therefore, we propose an extension of a previous positive-definite tensor glyph [28] to the full space of symmetric second-order tensors. Our glyph emphasizes differences in eigenvalue sign in a way that, unlike the Reynolds glyph [18], prevents small eigenvalues from being occluded by larger ones. We also propose to use halos to ensure tensor glyph visibility even when one or more eigenvalues are near zero. Finally, we present a time- and memory-efficient implementa-

2118 IEEE TRANSACTIONS ON VISUALIZATION AND COMPUTER GRAPHICS, VOL. 18, NO. 12, DECEMBER 2010

Table 1. Hypothesis tests used to reach conclusions for the dependent measures. Sharp results will notice that one subject's TLX ratings were discarded due to two less than the number of cases yielded by the study design, in which all factors were crossed for error, response time, and answers.

Test and Factor(s)	Error	Response Time	Number of Answers	Workload
ANOVA: MVV Technique	$F(3, 33) = 32.65, p = 0.00$	$F(3, 33) = 35.48, p = 0.00$	$F(3, 33) = 45.57, p = 0.00$	$F(3, 30) = 19.20, p = 0.00$
Host: DDS vs. Slivers	$t(106) = 6.59, p = 0.00$	$t(106) = 3.14, p = 0.002$	$t(106) = 0.65, p = 0.52$	$t(9) = 1.33, p = 0.22$
Host: DDS vs. Att. Blocks	$t(106) = 5.20, p = 0.00$	$t(106) = 3.23, p = 0.002$	$t(106) = 2.54, p = 0.01$	$t(9) = 1.05, p = 0.32$
ANOVA: Num. of Layers	$F(2, 22) = 7.45, p = 0.003$	$F(2, 22) = 5.37, p = 0.01$	$F(2, 22) = 1.80, p = 0.19$	$F(2, 22) = 4.30, p = 0.03$
ANOVA: Target Size	$F(2, 22) = 89.92, p = 0.00$	$F(2, 22) = 8.98, p = 0.001$	$F(2, 22) = 4.30, p = 0.03$	$F(1, 10) = 0.06, p = 0.82$
ANOVA: Experience	$F(1, 10) = 1.83, p = 0.21$	$F(1, 10) = 6.17, p = 0.03$	$F(1, 10) = 0.06, p = 0.82$	$F(1, 10) = 0.06, p = 0.82$
ANOVA: MVV-by-Num. Layers	$F(4, 44) = 8.79, p = 0.00$	$F(4, 44) = 9.06, p = 0.00$	$F(4, 44) = 3.15, p = 0.02$	$F(4, 44) = 3.15, p = 0.02$

Table 2. The mean and standard deviation for each MVV technique for each of the three objective dependent measures and the subjective workload rating shows the difficulty users had in attempting to complete the task with the baseline technique. Error is expressed in units of layers (range 0-6), time in seconds, answers in a count, and workload through NASA TLX.

Name	Error (layers)	Std. Dev.	Time (sec)	Time (sec)	Answers (count)	Answers (count)	Workload (TLX)	Std. Dev.
JuxLayers	1.54	1.03	46.72	36.44	8.87	8.09	65.53	15.40
DDSpots	0.09	0.21	7.33	6.01	1.09	0.28	40.74	23.17
Slivers	0.42	0.48	9.62	4.25	1.11	0.24	27.86	22.38
Attrib	0.47	0.73	9.74	5.27	1.20	0.37	31.79	16.24

Fig. 6. Graph of dependent measures for the integrated MVV techniques. Error (red) and workload (green), but not on time (blue) or answers selected (orange). Error and number changes to answers (i.e., one less than the number of answers selected to align the graphs better) are on the primary axis on the left. Response time and workload are on the secondary axis on the right.

3.5.5 Effect of User Experience

We expected (based on our past work) to see users who had participated in previous studies perform faster. We found a main effect of (binary) user experience on response time (Table 1). Returning users were on average 30.7% faster than subjects who were participating in our sequence of studies for the first time. This confirms Hypothesis 5.

3.5.6 Other Findings

We found significant interactions between MVV Technique and the number of target layers for error and for response time. We found significant interactions between MVV Technique and the target size for error and for response time. Since these results (Table 1) give us insight into the usability of the techniques and also implicitly show the main effects of the number of target layers and of the target size, we graph these results in Figure 7.

There was a significant interaction between the number of target layers and the target size for error  $-F(4, 44) = 5.128, p = 0.002$ . For all number of target layers, the smallest targets were most difficult, but the magnitude of the increase in difficulty from the middle size down to the smallest size was quite a bit lower for five layers than would be expected looking at the jumps for four and six layers.

We checked whether fatigue had an effect on error by running a 3 (MVV Technique)  $\times$  36 (Count) ANOVA with the MVV technique and the count of questions as factors; we found no significant effect of the count of questions completed  $-F(35, 385) = 0.798, p = 0.789$ . Similarly, we conducted a 3 (MVV Technique)  $\times$  3 (Target Layers)  $\times$  3 (Target Size)  $\times$  4 (Repetition) ANOVA to see if repetition of the combination of target size and number of target layers had a main effect; we found no significant effect  $-F(3, 33) = 0.860, p = 0.472$ . Analogous ANOVA calculations revealed that there was no significant effect of trial count or repetition on the number of answers selected.

We ran a filter on the error to find trials where the response was judged to be incorrect, but the error in pixels from the correct answer was smaller than the size of the target. There were only nine such errors in 494 trials that saw errors (out of 1728 total trials), so we cannot attach statistical significance to the occurrence of such an event. But we do find it curious to note that of the nine such errors, seven saw selections that were no more than seven pixels away from the target patch – and all of these were trials with Attribute Blocks and the smallest target size (31 pixels). (Two trials saw selections that were almost the size of the target patch – 61 or 91 pixels, respectively – with Oriented Slivers.)

3.5.4 Effect of Size of Target

The size of the target had a main effect on error, response time, and the number of answers selected. However, the results again do not support the complete ordering predicted in Hypothesis 4. The smallest target size was clearly more difficult, but there was no significant difference between the two larger sizes. Users were fastest with the largest target size, with a small but not significant difference between the smallest and middle sizes. Users changed their answers at a slightly increasing rate with decreasing target size.

Linstrom et al. [2006]

Schultz & Kindlmann [2010]

Livingston et al. [2012]

# Results: evaluation scenarios (selection)

algorithmic performance  
35% of scenarios

qualitative result inspection  
46% of scenarios

user performance/experience  
14% of scenarios

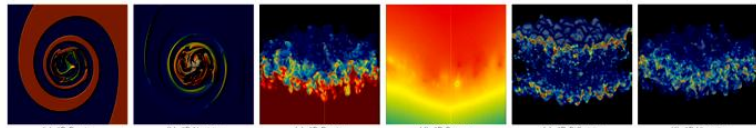


Fig. 1. Visualizations of 2D data (as pseudocolored height fields) and 3D data (volume rendered) used in our experiments.

data set	compressed size (MB) and compression time (seconds)					compressed size (MB) and compression time (seconds)											
	name	unique entropy range (%)	entropy (bits)	min	max	size (MB)	time (sec)	zlib	[RKB2006]	[EFF2000]	[LS2005]	new scheme					
m2d density	3.89	3.49	21.83	8.7E-01	1.2E+00	19.6	0.71	1.6	0.86	4.3	0.49	4.4	0.56	1.3	1.08	1.3	0.56
m2d vorticity	99.20	22.25	31.05	-1.4E+02	2.5E+01	19.6	0.71	18.4	2.14	11.8	1.21	15.5	1.29	12.9	2.22	13.8	1.49
m3d density	7.67	5.16	23.60	1.0E+00	3.0E+00	364.5	12.81	50.4	17.55	100.5	9.06	96.3	8.48	35.7	19.03	35.5	9.25
m3d pressure	27.29	23.91	31.06	-3.7E+00	2.3E+03	364.5	12.80	229.2	99.76	95.6	9.31	87.9	8.87	40.1	18.79	40.4	9.96
m3d diffusivity	36.87	23.19	30.02	0.0E+00	6.8E+00	364.5	12.68	297.6	42.90	250.8	19.09	239.3	15.02	198.8	31.92	203.0	18.47
m3d viscosity	50.07	24.86	28.59	8.6E-15	2.9E+00	364.5	12.62	314.0	46.09	249.4	18.95	246.1	14.68	209.2	32.66	207.5	19.45
h3d temp	65.70	23.54	31.56	-7.7E+01	1.0E+35	95.4	3.77	75.8	14.56	59.3	4.64	53.0	4.27	44.1	8.04	44.1	5.06
h3d pressure	81.82	24.13	31.58	-3.4E+03	1.0E+35	95.4	3.78	82.3	12.00	64.3	5.14	52.9	4.87	45.0	7.78	45.2	5.34
h3d x velocity	84.18	24.18	31.55	-5.3E+01	1.0E+35	95.4	3.89	86.1	11.27	67.4	6.22	63.3	4.59	54.5	8.86	55.4	5.44
h3d y velocity	84.32	24.18	31.55	-4.6E+01	1.0E+35	95.4	3.83	84.5	11.42	67.1	5.74	62.3	5.04	53.5	8.64	53.8	5.53
h3d z velocity	86.82	24.24	31.54	-3.2E+00	1.0E+35	95.4	3.87	88.4	10.76	85.6	8.50	76.9	5.29	68.9	9.83	69.1	6.65
M3d density	40.14	18.81	22.59	1.0E+00	3.0E+00	288.0	11.28	136.8	41.91	160.3	11.63	121.6	10.94	-	-	105.2	11.63
M3d pressure	100.00	25.17	63.00	-2.2E+00	2.2E+00	288.0	11.20	272.6	35.18	237.3	14.91	225.1	16.59	-	-	208.4	17.20
M3d x velocity	100.00	25.17	63.00	-2.2E+00	2.3E+00	288.0	10.83	275.6	32.30	230.4	14.73	215.1	15.91	-	-	197.7	16.84
M3d y velocity	100.00	25.17	63.00	-2.1E+00	2.3E+00	288.0	10.54	275.1	32.19	223.1	14.27	215.2	15.16	-	-	197.7	16.65
M3d z velocity	100.00	25.17	63.00	-5.2E+00	9.0E+00	288.0	10.32	275.5	32.62	226.6	14.74	213.7	16.05	-	-	196.8	16.14
atom x position	61.10	23.82	31.01	-4.8E-02	4.6E+02	107.7	7.07	84.3	21.18	76.0	7.88	78.8	7.61	67.3	12.88	68.6	9.07
atom y position	45.90	23.32	26.99	3.7E-02	2.1E+03	107.7	7.08	65.9	30.76	60.4	6.97	56.4	6.31	47.0	10.49	46.9	7.73
atom z position	61.68	23.84	27.48	9.1E-05	4.6E+02	107.7	7.46	94.6	19.86	82.6	9.00	86.1	8.25	75.7	13.80	78.2	9.93
atom x velocity	64.65	23.87	30.96	1.5E-01	1.4E-01	107.7	7.30	95.7	19.88	93.8	10.07	90.1	9.65	84.8	14.93	87.6	9.92
atom temp	64.91	23.94	27.41	3.0E-03	7.1E+03	107.7	6.69	95.7	19.76	91.6	10.27	95.9	8.34	84.6	15.02	84.6	10.31
atom energy	3.45	18.57	21.79	-3.6E+00	-2.7E+00	107.7	7.15	77.9	38.59	74.1	7.98	71.8	7.01	60.8	12.66	60.5	8.30
lucy	61.39	24.38	31.09	-6.1E+02	1.2E+03	160.5	-	137.8	-	99.5	-	90.0	-	73.6	-	77.8	-
david10mm	25.23	17.08	31.11	-4.4E+03	1.8E+03	322.5	-	144.9	-	155.7	-	163.4	-	108.6	-	131.9	-
torso	84.72	18.48	31.08	-2.7E+02	5.8E+02	1.9	-	1.7	-	1.5	-	1.5	-	1.3	-	1.3	-
rhl	71.90	20.14	25.99	1.5E+00	3.6E+02	8.4	-	7.1	-	5.8	-	5.6	-	4.7	-	4.8	-

Table 1. Compression results for the Miranda (m2d, m3d, M3d) structured grids, the atom point set, the lucy and david triangle meshes, and the torso and rhl tetrahedral meshes. All data but M3d is represented in single precision. The [LS2005] scheme operates on single precision only, hence the missing values. For the meshes we report only the compressed size of vertex coordinates, timings are dominated by connectivity coding, and are hence excluded. The range measures (the logarithm of) the number of floating-point values between min and max. Note that the first-order entropy is limited by the number of samples in a data set.

tion.) Arguably such data sets should use an integer rather than floating-point representation, although for simplicity or other reasons it is common practice to use floating-point. Contrary to [16], which entropy codes all bits of the residual, our new coder sacrifices such potential compression gains for speed by storing these repeated low-order bits in raw and uncompressed form. However, the massive data sets from scientific simulation that motivated our work on high-speed compression, as well as our tetrahedral meshes, rarely exhibit significant low-order redundancy, as also evidenced by our results.

### 5.1.1 Lossy Compression

Fig. 3 shows that our scheme gracefully adapts to decreasing levels of precision when discarding the least significant mantissa (and eventually exponent) bits. For  $n$  bits of precision, the schemes [7, 22] require  $\log_2 n$  bits to code the number of leading zeros, whereas our scheme exploits the combination of low entropy in the leading-zero count and the elimination of the low-order bits that are most difficult to predict and compress.

### 5.2 Compression Speed

Fig. 4 shows the speed of compressing from memory to disk, including disk write time. Because the throughput is only 20% less than our method, its decompression speed is similar to its compression speed.) We also include the raw I/O performance of dumping the data uncompressed using a single `fwrite` call. Timings

correspond to the median of five runs. Whereas our compressor is slightly slower than the less effective compressors [7, 22], it is nearly twice as fast as [16] while producing similar compression rates. However, in more I/O-intensive scenarios, such as in massively parallel simulations dumping data to the same file system (as is common), the improved compression of our method over [7, 22] results in a net gain in effective throughput. We integrated our compression code with Miranda's dump routines and ran performance tests on 276 nodes of LLNL's ASC supercomputer. Achieving on average a lossless reduction of 3.7 on 75 GB of data dumped, the overall dump time was reduced by a factor of 2.7 over writing the data uncompressed.

### 5.3 Entropy Coding

We compared the raw throughput of our range coder and Schindler's [23] by (1) passing raw bytes through it with no compression and (2) entropy coding byte sequences. In both cases, the source data is the uncompressed floating-point data used in our experiments. Timings show that our coder is 40% faster for raw transmission and 28% faster for entropy coding. Meanwhile, the inefficiency of our coder due to loss of precision and range reduction is only 26 bytes of overhead for 1.5 GB of coded data. Its raw throughput is only 20% less than an `fwrite` call, while its entropy coding throughput of 20 MB per second, which includes probability modeling and I/O time, compares favorably with state-of-the-art entropy coders [25].

Linstrom et al. [2006]

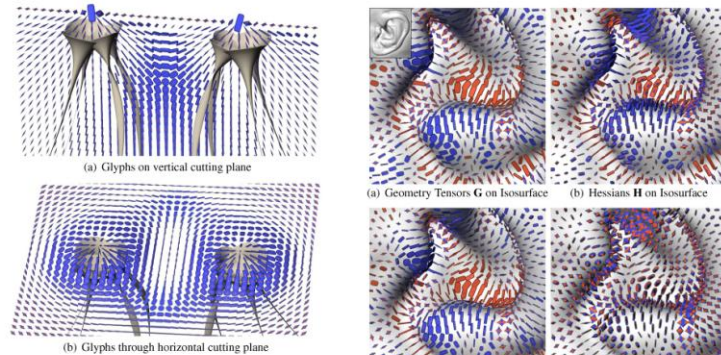


Fig. 8. Glyphs in the double point load stress tensor field reveal the minor eigenvector along which hyperstreamlines [7] are traced (a), and the variation in stress with distance from the load (b).

volume rendered [32], but its eigenvectors are commonly used in non-photo-realistic rendering, e.g. curvature-based strokes [11, 14, 19]. Inspecting geometry tensors could help debug an NPR method giving unexpected results in an unfamiliar dataset. Fig. 9(a) visualizes geometry tensors  $G$  on an isosurface (sampled by a particle system [39]) of an ear from the Visible Human male CT scan. Variations in surface curvature are reflected in the new glyphs: convex (blue circles), concave (orange circles), and saddles (orange and blue stars). For comparison, Fig. 9(b) shows the full Hessian  $H$  from which  $G$  was computed.

The new glyphs may also have a role in visualizing the tensor ingredients of image analysis methods such as edge detection. One edge definition is zero-crossing on the second directional derivative along the gradient direction,  $f'' = \mathbf{n}^T H \mathbf{n}$ . This edge surface is sampled by a particle system [33] in Fig. 9(c), showing the Hessians at the edge locations, and revealing close similarities with the geometry tensors on the isosurface in Fig. 9(a), indicating that one of the Hessian eigenvalues is near zero even though this is not part of the edge definition. Another edge definition is the zero-crossing of the Laplacian  $\nabla^2 f = \text{tr}(H)$ , and Fig. 9(d) illustrates the difference between the Hessians on this surface and those in Fig. 9(c). The consistently gray glyph halos in Fig. 9(d) indicate that these are traceless tensors.

As a demonstration of the glyphs in a 2-D visualization, Fig. 10 visualizes a cross-section of a simulation of jet flow rightward into a steady medium, causing turbulence. Glyphs of rate-of-deformations tensors document how an infinitesimal volume is stretched or compressed as it moves along the flow. A backdrop of line integral convolution [4] (with contrast modulated by velocity) provides visual context. Fig. 10(a) uses the exponentially-scaled ellipses of [34] to map tensors with negative eigenvalues to positive-definite tensors suitable for ellipsoid visualization. When the absolute difference between eigenvalues becomes too large, these glyphs can become so stretched that they overlap each other and extend over a significant portion of the domain, undermining the locality normally enjoyed by glyphs. Such stretching also reduces the visual presence of the needle-like glyphs for tensors with larger norms, contrary to scale preservation (6). Fig. 10(b) uses our superquadratic glyphs with  $\sigma(\|D\|) \approx \|D\|$ . The aspect ratio reflects the relative eigenvalue magnitudes, the size correctly indicates the tensor norm, and pointed glyphs shapes clearly communicate eigenvector directions. With compression of scale variation

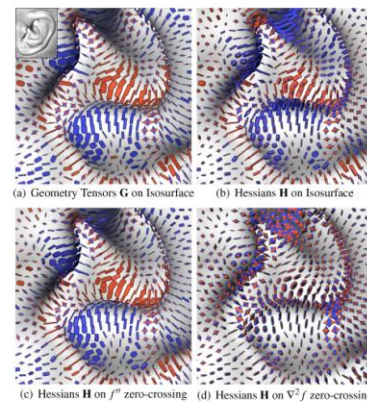


Fig. 9. Visualization of geometry tensors (a) and Hessians associated with isosurfaces (b) and two different definitions of edges, zero-crossings of the second-directional derivative (c) and the Laplacian (d). These results use  $\sigma(\|D\|) \approx \|D\|^{1/2}$  in (6).

$\sigma(\|D\|) \approx \|D\|^{1/2}$ , Fig. 10(c) better shows the directional patterns where the tensor norm is low. Colormapping the rate-of-deformation tensor trace with glyph halos highlights the regions of over-all stretching or compression, especially along the bottom edge of the domain.

Finally, Fig. 11 demonstrates how our new glyph performs traceless tensor visualization, in a side-by-side comparison to the dedicated traceless NLG tensor glyphs by Jankun-Kelly et al. [25]. Traceless tensors form a plane in eigenvalue space, and we are visualizing samples from a square within this plane, centered around the zero tensor (cf. Fig. 4(c)). Unlike the traceless glyph, which maps tensor norm to glyph sharpness, our glyph expresses norm by its overall scale  $\sigma(\|D\|) \approx \|D\|$ . Consequently, the traceless glyph requires pre-specification of maximum eigenvalue magnitudes (which are mapped to perfect sharpness), while our glyph can be used without such prior information. Another notable difference is that limiting their glyph to traceless tensors allows Jankun-Kelly et al. to make use of parts of the superquadratic shape space - including cylinders and boxes - that our approach sets aside for positive- or negative-definite tensors.

### 6 CONCLUSION

Visualization research has made significant progress in visualizing second-order tensor fields, but has mostly concentrated on the positive-definite case. Faced with indefinite tensors, a frequent strategy is to map them to positive-definite tensors prior to visualization [34, 22, 21, 52, 33]. Even when bijective mappings are used (so mathematically, no information is lost), such mappings still visually obscure the difference between positive and negative eigenvalues, which is a fundamental qualitative aspect in various applications.

Therefore, we propose an extension of a previous positive-definite tensor glyph [28] to the full space of symmetric second-order tensors. Our glyph emphasizes differences in eigenvalue sign in a way that, unlike the Reynolds glyph [18], prevents small eigenvalues from being occluded by larger ones. We also propose to use halos to ensure tensor glyph visibility even when one or more eigenvalues are near zero. Finally, we present a time- and memory-efficient implementa-

Schultz & Kindlmann [2010]

Table 1. Hypothesis tests used to reach conclusions for the dependent measures. Sharp results will notice that one subject's TLX ratings were discarded due to two less than the number of cases yielded by the study design, in which all factors were crossed for error, response time, and answers.

Test and Factor(s)	Error	Response Time	Number of Answers	Workload
ANOVA: MVV Technique	$F(3, 33) = 32.65, p = 0.00$	$F(3, 33) = 35.48, p = 0.00$	$F(3, 33) = 45.57, p = 0.00$	$F(3, 30) = 19.20, p = 0.00$
Post: DDS vs. Slivers	$t(106) = 6.59, p = 0.00$	$t(106) = 3.14, p = 0.002$	$t(106) = 0.65, p = 0.52$	$t(9) = 1.33, p = 0.22$
Post: DDS vs. Att. Blocks	$t(106) = 5.20, p = 0.00$	$t(106) = 3.23, p = 0.002$	$t(106) = 2.54, p = 0.01$	$t(9) = 1.05, p = 0.32$
ANOVA: Num. of Layers	$F(2, 22) = 7.45, p = 0.003$	$F(2, 22) = 5.37, p = 0.01$	$F(2, 22) = 1.80, p = 0.19$	$F(2, 22) = 4.30, p = 0.03$
ANOVA: Target Size	$F(2, 22) = 89.92, p = 0.00$	$F(2, 22) = 8.98, p = 0.001$	$F(2, 22) = 4.30, p = 0.03$	$F(1, 10) = 0.06, p = 0.82$
ANOVA: Experience	$F(1, 10) = 1.83, p = 0.21$	$F(1, 10) = 6.17, p = 0.03$	$F(1, 10) = 0.06, p = 0.82$	-
ANOVA: MVV-by-Num. Layers	$F(4, 44) = 8.79, p = 0.00$	$F(4, 44) = 9.06, p = 0.00$	$F(4, 44) = 3.15, p = 0.02$	-
ANOVA: MVV-by-Target Size	$F(4, 44) = 35.64, p = 0.00$	$F(4, 44) = 3.15, p = 0.02$	-	-

Table 2. The mean and standard deviation for each MVV technique for each of the three objective dependent measures and the subjective workload rating shows the difficulty users had in attempting to complete the task with the baseline technique. Error is expressed in units of layers (range 0-6), time in seconds, answers in a count, and workload through NASA TLX.

Name	Error (layers)	Std. Dev.	Time (sec)	Time (sec)	Answers (count)	Answers (Std. Dev.)	Workload (TLX)	Std. Dev.
JuxLayers	1.54	1.03	46.72	36.44	8.87	8.09	65.53	15.40
DDSpots	0.09	0.21	7.33	6.01	1.09	0.28	40.74	23.17
Slivers	0.42	0.48	9.62	4.25	1.11	0.24	27.86	22.38
Attrib	0.47	0.73	9.74	5.27	1.20	0.37	31.79	16.24

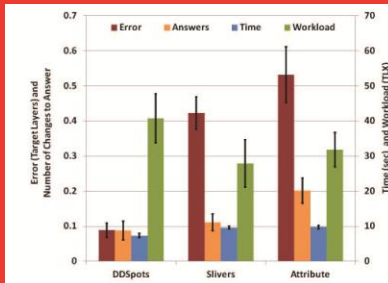


Fig. 6. Graph of dependent measures for the integrated MVV techniques. There was a main effect on error (red) and workload (green), but not on time (blue) or answers selected (orange). Error and number of changes to answers (i.e., one less than the number of answers selected to align the graphs better) are on the primary axis on the left. Response time and workload are on the secondary axis on the right.

that stated only that six layers would be the most difficult and not predicted a complete ordering with respect to increasing number of layers. Further, we found that users were 13% faster with six layers than the target than with five layers, which is a bit counter-intuitive and result that we shall discuss in Section 4.

### 3.5.4 Effect of Size of Target

The size of the target had a main effect on error, response time, and the number of answers selected. However, the results again do not support the complete ordering predicted in Hypothesis 4. The smallest target size was clearly more difficult, but there was no significant difference between the two larger sizes. Users were fastest with the largest target size, with a small but not significant difference between the smallest and middle sizes. Users changed their answers at a slightly increasing rate with decreasing target size.

### 3.5.5 Effect of User Experience

We expected (based on our past work) to see users who had participated in previous studies perform faster. We found a main effect of (binary) user experience on response time (Table 1). Returning users were on average 30.7% faster than subjects who were participating in our sequence of studies for the first time. This confirms Hypothesis 5.

### 3.5.6 Other Findings

We found significant interactions between MVV Technique and the number of target layers for error and for response time. We found significant interactions between MVV Technique and the target size for error and for response time. Since these results (Table 1) give us insight into the usability of the techniques and also implicitly show the main effects of the number of target layers and of the target size, we graph these results in Figure 7.

There was a significant interaction between the number of target layers and the target size for error -  $F(4, 44) = 5.128, p = 0.002$ . For all number of target layers, the smallest targets were most difficult, but the magnitude of the increase in difficulty from the middle size down to the smallest size was quite a bit lower for five layers than would be expected looking at the jumps for four and six layers.

We checked whether fatigue had an effect on error by running a (MVV Technique)  $\times$  36 (Count) ANOVA with the MVV technique and the count of questions as factors; we found no significant effect of the count of questions completed -  $F(35, 385) = 0.798, p = 0.78$ . Similarly, we conducted a 3 (MVV Technique)  $\times$  3 (Target Layer)  $\times$  3 (Target Size)  $\times$  4 (Repetition) ANOVA to see if repetition of the combination of target size and number of target layers had a main effect; we found no significant effect -  $F(3, 33) = 0.860, p = 0.47$ . Analogous ANOVA calculations revealed that there was no significant effect of trial count or repetition on the number of answers selected.

We ran a filter on the error to find trials where the response was judged to be incorrect, but the error in pixels from the correct answer was smaller than the size of the target. There were only nine such errors in 494 trials that saw errors (out of 1728 total trials), so we cannot attach statistical significance to the occurrence of such an event. But we do find it curious to note that of the nine such errors, seven saw selections that were no more than seven pixels away from the target patch - and all of these were trials with Attribute Blocks and the smallest target size (31 pixels). (Two trials saw selections that were almost the size of the target patch - 61 or 91 pixels, respectively with Oriented Slivers.)

Livingston et al. [2012]

# Results: evaluation scenarios (selection)

algorithmic performance  
35% of scenarios

qualitative result inspection  
46% of scenarios

user performance/experience  
14% of scenarios

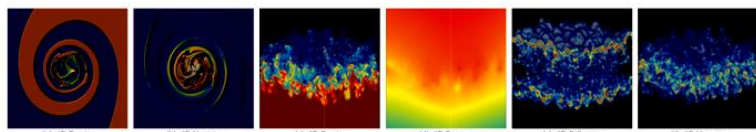


Fig. 1. Visualizations of 2D data (as pseudocolored height fields) and 3D data (volume rendered) used in our experiments.

name	data set				compressed size (MB) and compression time (seconds)											
	unique entropy (%)	range (bits)	min	max	size (MB)	time (sec)	zlib	[RKB2006]	[EFF2000]	[ILS2005]	new scheme	zlib	[RKB2006]	[EFF2000]	[ILS2005]	new scheme
m2d density	3.89	3.49 21.83	8.7E-01	1.2E+00	19.6	0.71	1.6	0.86	4.3	0.49	4.4	0.56	1.3	1.08	1.3	0.56
m2d vorticity	99.20	22.25 31.05	-1.4E+02	2.5E+01	19.6	0.71	18.4	2.14	11.8	1.21	15.5	1.29	12.9	2.22	13.8	1.49
m3d density	7.67	5.16 23.60	1.0E+00	3.0E+00	364.5	12.81	50.4	17.55	100.5	9.06	96.3	8.48	35.7	19.03	35.5	9.25
m3d pressure	27.29	23.91 31.06	-3.7E+00	2.3E+03	364.5	12.80	229.2	99.76	95.6	9.31	87.9	8.87	40.1	18.79	40.4	9.96
m3d diffusivity	36.87	23.19 30.02	0.0E+00	6.8E+00	364.5	12.68	297.6	42.90	250.8	19.09	239.3	15.02	198.8	31.92	203.0	18.47
m3d viscosity	50.07	24.86 28.59	8.6E-15	2.9E+00	364.5	12.62	314.0	46.09	249.4	18.95	246.1	14.68	209.2	32.66	207.5	19.45
h3d temp	65.70	23.54 31.56	-7.7E+01	1.0E+35	95.4	3.77	75.8	14.56	59.3	4.64	53.0	4.27	44.1	8.04	44.1	5.06
h3d pressure	81.82	24.13 31.58	-3.4E+03	1.0E+35	95.4	3.78	82.3	12.00	64.3	5.14	52.9	4.87	45.0	7.78	45.2	5.34
h3d x velocity	84.18	24.18 31.55	-5.3E+01	1.0E+35	95.4	3.89	86.1	11.27	67.4	6.22	63.3	4.59	54.5	8.86	53.4	5.44
h3d y velocity	84.32	24.18 31.55	-4.6E+01	1.0E+35	95.4	3.83	84.5	11.42	67.1	5.74	62.3	5.04	53.5	8.64	53.8	5.53
h3d z velocity	86.82	24.24 31.54	-3.2E+00	1.0E+35	95.4	3.87	88.4	10.76	85.6	8.50	76.9	5.29	68.9	9.83	69.1	6.65
M3d density	40.14	18.81 22.59	1.0E+00	3.0E+00	288.0	11.28	136.8	41.91	160.3	11.63	121.6	10.94	-	-	105.2	11.63
M3d pressure	100.00	25.17 63.00	-2.2E+00	2.2E+00	288.0	11.20	272.6	35.18	237.3	14.91	225.1	16.59	-	-	208.4	17.20
M3d x velocity	100.00	25.17 63.00	-2.2E+00	2.3E+00	288.0	10.83	275.6	32.30	230.4	14.73	215.1	15.91	-	-	197.7	16.84
M3d y velocity	100.00	25.17 63.00	-2.1E+00	2.3E+00	288.0	10.54	275.1	32.19	223.1	14.27	215.2	15.16	-	-	197.7	16.65
M3d z velocity	100.00	25.17 63.00	-5.2E+00	9.0E+00	288.0	10.32	275.5	32.62	226.6	14.74	213.7	16.05	-	-	196.8	16.14
atom x position	61.10	23.82 31.01	-4.8E-02	4.6E+02	107.7	7.07	84.3	21.18	76.0	7.88	78.8	7.61	67.3	12.88	68.6	9.07
atom y position	45.90	23.32 26.99	3.7E-02	2.1E+03	107.7	7.08	65.9	30.76	60.4	6.97	56.4	6.31	47.0	10.49	46.9	7.73
atom z position	61.68	23.84 27.48	9.1E-05	4.6E+02	107.7	7.46	94.6	19.86	82.6	9.00	86.1	8.25	75.7	13.80	78.2	9.93
atom y velocity	64.65	23.87 30.96	1.5E-01	1.4E-01	107.7	7.30	95.7	19.88	93.8	10.07	99.1	9.65	84.8	14.93	87.6	9.92
atom temp	64.91	23.94 27.41	3.0E-03	7.1E+03	107.7	6.69	95.7	19.76	91.6	10.27	95.9	8.34	84.6	15.02	84.6	10.31
atom energy	3.45	18.57 21.79	-3.6E+00	-2.7E+00	107.7	7.15	77.9	38.59	74.1	7.98	71.8	7.01	60.8	12.66	60.5	8.30
lucy	61.39	24.38 31.09	-6.1E-02	1.2E+03	160.5	-	137.8	-	99.5	-	90.0	-	73.6	-	77.8	-
david1 <sub>norm</sub>	25.23	17.08 31.11	-4.4E+03	1.8E+03	322.5	-	144.9	-	155.7	-	163.4	-	108.6	-	131.9	-
torso	84.72	18.48 31.08	-2.7E+02	5.8E+02	1.9	-	1.7	-	1.5	-	1.5	-	1.3	-	1.3	-
tbl	71.90	20.14 25.99	1.5E+00	3.6E+02	8.4	-	7.1	-	5.8	-	5.6	-	4.7	-	4.8	-

Table 1. Compression results for the Miranda (m2d, m3d, M3d) structured grids, the atom point set, the lucy and david triangle meshes, and the torso and tbl tetrahedral meshes. All data but M3d is represented in single precision. The [ILS2005] scheme operates on single precision only, hence the missing values. For the meshes we report only the compressed size of vertex coordinates; timings are dominated by connectivity coding, and are hence excluded. The range measures (the logarithm of) the number of floating-point values between min and max. Note that the first-order entropy is limited by the number of samples in a data set.

tation.) Arguably such data sets should use an integer rather than floating-point representation, although for simplicity or other reasons it is common practice to use floating-point. Contrary to [16], which entropy codes all bits of the residual, our new coder sacrifices such potential compression gains for speed by storing these repeated low-order bits in raw and uncompressed form. However, the massive data sets from scientific simulation that motivated our work on high-speed compression, as well as our tetrahedral meshes, rarely exhibit any significant low-order redundancy, as also evidenced by our results.

### 5.1.1 Lossy Compression

Fig. 3 shows that our scheme gracefully adapts to decreasing levels of precision when discarding the least significant mantissa (and eventually exponent) bits. For  $n$  bits of precision, the schemes [7, 22] require  $\log_2 n$  bits to code the number of leading zeros, whereas our scheme exploits the combination of low entropy in the leading-zero count and the elimination of the low-order bits that are most difficult to predict and compress.

### 5.2 Compression Speed

Fig. 4 shows the speed of compressing from memory to disk, including disk write time. Because the throughput is only 20% less than our method, its decompression speed is similar to its compression speed.) We also include the raw I/O performance of dumping the data uncompressed using a single `fwrite` call. Timings

correspond to the median of five runs. Whereas our compressor is slightly slower than the less effective compressors [7, 22], it is nearly twice as fast as [16] while producing similar compression rates. However, in more I/O-intensive scenarios, such as in massively parallel simulations dumping data to the same file system (as is common), the improved compression of our method over [7, 22] results in a net gain in effective throughput. We integrated our compression code with Miranda's dump routines, and ran performance tests on 276 nodes of LLNL's MPP supercomputer. Achieving on average a lossless reduction of 3.7 on 75 GB of data dumped, the overall dump time was reduced by a factor of 2.7 over writing the data uncompressed.

### 5.3 Entropy Coding

We compared the raw throughput of our range coder and Schindler's [23] by (1) passing raw bytes through it with no compression and (2) entropy coding byte sequences. In both cases, the source data was the uncompressed floating-point data used in our experiments. Timings show that our coder is 40% faster for raw transmission and 28% faster for entropy coding. Meanwhile, the inefficiency of our coder due to loss of precision and range reduction is only 26 bytes of overhead for 1.5 GB of coded data. Its raw throughput is only 20% less than an `fwrite` call, while its entropy coding throughput of 20 MB per second, which includes probability modeling and I/O time, compares favorably with state-of-the-art entropy coders [25].

Lindström et al. [2006]

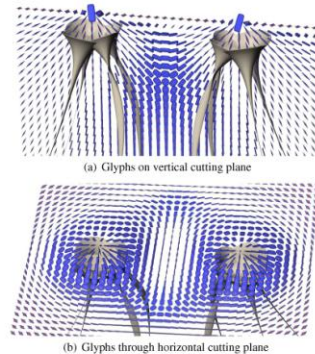


Fig. 8. Glyphs in the double point load stress tensor field reveal the minor eigenvector along which hyperstreamlines [7] are traced (a), and the variation in stress with distance from the load (b).

volume rendered [32], but its eigenvectors are commonly used in non-photo-realistic rendering, e.g. curvature-based strokes [11, 14, 19]. Inspecting geometry tensors could help debug an NPR method giving unexpected results in an unfamiliar dataset. Fig. 9(a) visualizes geometry tensors  $G$  on an isosurface (sampled by a particle system [39]) of an ear from the Visible Human male CT scan. Variations in surface curvature are reflected in the new glyphs: convex (blue circles), concave (orange circles), and saddles (orange and blue stars). For comparison, Fig. 9(b) shows the full Hessian  $H$  from which  $G$  was computed.

The new glyphs may also have a role in visualizing the tensor ingredients of image analysis methods such as edge detection. One edge definition is zero-crossing on the second directional derivative along the gradient direction,  $f'' = \mathbf{n}^T H \mathbf{n}$ . This edge surface is sampled by a particle system [33] in Fig. 9(c), showing the Hessians at the edge locations, and revealing close similarities with the geometry tensors on the isosurface in Fig. 9(a), indicating that one of the Hessian eigenvalues is near zero even though this is not part of the edge definition. Another edge definition is the zero-crossing of the Laplacian  $\nabla^2 f = \text{tr}(H)$ , and Fig. 9(d) illustrates the difference between the Hessians on this surface and those in Fig. 9(c). The consistently gray glyph halos in Fig. 9(d) indicate that these are traceless tensors.

As a demonstration of the glyphs in a 2-D visualization, Fig. 10 visualizes a cross-section of a simulation of jet flow rightward into a steady medium, causing turbulence. Glyphs of rate-of-deformations tensors document how an infinitesimal volume is stretched or compressed as it moves along the flow. A backdrop of line integral convolution [4] (with contrast modulated by velocity) provides visual context. Fig. 10(a) uses the exponentially-scaled ellipses of [34] to map tensors with negative eigenvalues to positive-definite tensors suitable for ellipsoid visualization. When the absolute difference between eigenvalues becomes too large, these glyphs can become so stretched that they overlap each other and extend over a significant portion of the domain, undermining the locality normally enjoyed by glyphs. Such stretching also reduces the visual presence of the needle-like glyphs for tensors with larger norms, contrary to scale preservation (6). Fig. 10(b) uses our superquadratic glyphs with  $s(\|D\|) \approx \|D\|$ . The aspect ratio reflects the relative eigenvalue magnitudes, the size correctly indicates the tensor norm, and pointed glyph shapes clearly communicate eigenvector directions. With compression of scale variation

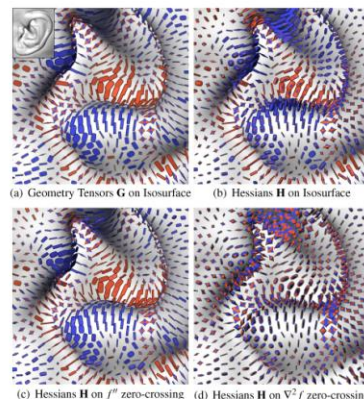


Fig. 9. Visualization of geometry tensors (a) and Hessians associated with isosurfaces (b) and two different definitions of edges, zero-crossings of the second-directional derivative (c) and the Laplacian (d). These results use  $s(\|D\|) \approx \|D\|^{1/2}$  in (6).

$s(\|D\|) \approx \|D\|^{1/2}$ , Fig. 10(c) better shows the directional patterns where the tensor norm is low. Colormapping the rate-of-deformation tensor trace with glyph halos highlights the regions of over-all stretching or compression, especially along the bottom edge of the domain.

Finally, Fig. 11 demonstrates how our new glyph performs traceless tensor visualization, in a side-by-side comparison to the dedicated traceless NLCT tensor glyphs by Jankun-Kelly et al. [25]. Traceless tensors form a plane in eigenvalue space, and we are visualizing samples from a square within this plane, centered around the zero tensor (cf. Fig. 4(c)). Unlike the traceless glyph, which maps tensor norm to glyph sharpness, our glyph expresses norm by its overall scale  $s(\|D\|) \approx \|D\|$ . Consequently, the traceless glyph requires pre-specification of maximum eigenvalue magnitudes (which are mapped to perfect sharpness), while our glyph can be used without such prior information. Another notable difference is that limiting their glyph to traceless tensors allows Jankun-Kelly et al. to make use of parts of the superquadratic shape space - including cylinders and boxes - that our approach sets aside for positive- or negative-definite tensors.

### 6 CONCLUSION

Visualization research has made significant progress in visualizing second-order tensor fields, but has mostly concentrated on the positive-definite case. Faced with indefinite tensors, a frequent strategy is to map them to positive-definite tensors prior to visualization [34, 22, 21, 52, 33]. Even when bijective mappings are used (so mathematically, no information is lost), such mappings still visually obscure the difference between positive and negative eigenvalues, which is a fundamental qualitative aspect in various applications.

Therefore, we propose an extension of a previous positive-definite tensor glyph [28] to the full space of symmetric second-order tensors. Our glyph emphasizes differences in eigenvalue sign in a way that, unlike the Reynolds glyph [18], prevents small eigenvalues from being occluded by larger ones. We also propose to use halos to ensure correct indication of the tensor norm, and pointed glyph shapes clearly communicate eigenvector directions. With compression of scale variation

Schultz & Kindlmann [2010]

two cases was not a factor that could affect the outcome of this study negatively.

One experimenter engaged in the demonstration of the table and in assisting the participant when needed. A second experimenter took notes and also documented the session by voice recording. A prepared interview guide was used. It included a set of predefined questions that covered various aspects of the design goals (section 5.1) and also a number of potential questions used to prompt the participant to "think aloud" when needed. Both experimenters engaged in the conversation and made sure that all questions in the interview guide were covered by the end of the session. Some questions were discussed while participants worked on the cases and some were reviewed afterwards.

After using the table participants completed a subjective satisfaction questionnaire. The responses were given on a 5-point rating scale: *Strongly unfavorable* (1), *Unfavorable* (2), *Unsure* (3), *Favorable* (4), and *Strongly favorable* (5). The questionnaire covered the following issues:

1. **Overall impression:** The overall impression of the table.
2. **Willingness to use:** Whether the orthopedic surgeon would like to use the table in their daily work.
3. **Efficiency for planning:** Whether using the table would save time for pre-operative planning.
4. **Efficiency for surgery:** Whether using the table pre-operatively would save time during actual surgery.
5. **Quality of work:** Whether using the table would improve the quality of clinical work.
6. **Learnability:** Ease of learning table usage for a novice user.
7. **Interaction:** Ease of interacting with the table.
8. **Benefits over static imaging:** Whether interactive 3D imaging is superior to the series of static 3D snap-shots used today.
9. **Collaboration:** Whether access to the table at work would facilitate collaboration between several people.
10. **Similarity to real situation:** Whether the similarity to a real situation (patient lying on a table) facilitates insights and decision making.
11. **Recommend to others:** Whether the orthopedic surgeon would recommend colleagues to use the table at work.

Please note that while the list above well represents the statements rated by the participants, the wording has been translated and slightly changed to clarify reporting of the results. A full session lasted for approximately 50 minutes including all parts.

### 7 RESULTS

The user study proved effective for the objective of collecting distinct and broad feedback from the orthopedic surgeons about how the visualization table would fit in their application domain. They did not consider the "think aloud" approach to be distracting from the evaluation tasks. The overall assessment from the surgeons is that the table would be useful in their clinical work. This is illustrated by the numerical ratings in the post-session questionnaire, see figure 12. Responses for the eleven statements has a group mean value<sup>1</sup> of 3.8 (two statements), 4 (two statements) and above 4 (seven statements) respectively, all corresponding to a clearly favorable rating. Statistical significance was, however, not achieved but this is to be expected for this small study. There is only one example of a negative rating, one surgeon expressed moderate disagreement with the table's potential to improve efficiency during surgery (specialist, age 50). There were three statements concerning general impressions of usefulness, *Overall impression*, *Willingness to use*, and *Recommend to others*, and in all three cases the study shows a strongly favorable mean rating of 4.4. Both the younger and less experienced participants and the older specialists contributed

<sup>1</sup>It can be discussed whether averaging in an ordinal scale is appropriate, our conclusion is that in this case it is the best way to convey the results, in combination with the min-max measures in figure 12.

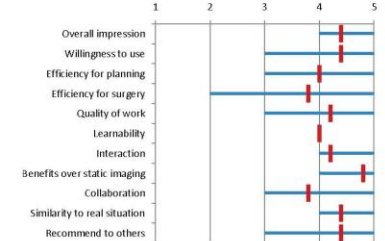


Fig. 12. The quantitative results of the user study questionnaire. Subjective satisfaction regarding use of the table was measured for 11 questions, see section 6. The 5-point rating scale ranges from *Strongly unfavorable* (1) through *Unsure* (3) to *Strongly favorable* (5). Vertical red bars denote the mean value and horizontal blue lines denote the full span of given ratings.

to all levels of the rating scale (3-5). Hence, for this participant pool, age and level of experience did not seem to affect the attitude towards the technique.

The issues behind the rather general statements in the questionnaire were discussed in greater detail during the sessions at the table and these findings provide a more nuanced and informative view of opinions. Below, these findings are summarized under the following four subheadings: Ease of use and learnability, Clinical usefulness, Work-flow, and Desired features.

### 7.1 Ease of use and learnability

Low learning threshold and high usability were central objectives in the design of the system, reflected by design requirements R1-R5. Regarding the overall impression of the table all participants expressed positive statements. The interface was considered intuitive and convenient, and it was easy to learn how to use the basic functionality. The comments about learning threshold expressed an anticipation that novice users would quickly learn the basic functionality, although some of the more advanced functionality (activated via the pucks) would require some practice. All appreciated the clean interface with only a few visible GUI elements and emphasized the benefit and importance of the screen being focused towards visualizing the 3D image.

Regarding the interaction, the touch gestures were described as intuitive and straightforward to use, also for one of the participants who pointed out that he had never used a touch-controlled interface before (specialist, age 54). The surgeons were asked if they perceived the interaction as robust and responsive. They all concurred, through statements that the result of actions on the screen was what they expected and that they felt in control. Nobody mentioned that the level of precision provided by the touch technology and the RST interaction was insufficient or problematic. Even though the participants did not bring it up, the experimenters noted a few occasions of unintentional gestures due to holding the knuckles of inactive fingers too close to the surface. The typical effect was that panning occurred instead of an intended x-y-rotation, which the users dealt with by lifting the hand and reapplying the rotation gesture. For the additional MPR slice views it was commented that touch gestures were more efficient for transversal browsing than using a mouse.

The pucks were, in general, described as a convenient approach for reaching additional features. The interaction that caused some confusion was the advanced parts of the clip plane functionality, namely to control and understand slab clipping. The surgeons adopted the natural size zoom as an integral part of the toolset and no usability obstacles

Lundström et al. [2011]

# Results: evaluation scenarios (selection)

algorithmic performance  
35% of scenarios

qualitative result inspection  
46% of scenarios

user performance/experience  
14% of scenarios

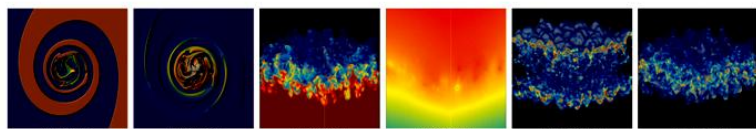


Fig. 1. Visualizations of 2D data (as pseudocolored height fields) and 3D data (volume rendered) used in our experiments.

name	data set			compressed size (MB) and compression time (seconds)													
	unique (%)	entropy (bits)	range (bits)	min	max	size (MB)	time (sec)	zlib	[RKB2006]	[EFF2000]	[ILS2005]	new scheme					
m2d density	3.89	3.49	21.83	8.7E-01	1.2E+00	19.6	0.71	1.6	0.86	4.3	0.49	4.4	0.56	1.3	1.08	1.3	0.56
m2d vorticity	99.20	22.25	31.05	-1.4E+02	2.5E+01	19.6	0.71	18.4	2.14	11.8	1.21	15.5	1.29	12.9	2.22	13.8	1.49
m3d density	7.67	5.16	23.60	1.0E+00	3.0E+00	364.5	12.81	50.4	17.55	100.5	9.06	96.3	8.48	35.7	19.03	35.5	9.25
m3d pressure	27.29	23.91	31.06	-3.7E+00	2.3E+03	364.5	12.80	229.2	99.76	95.6	9.31	87.9	8.87	40.1	18.79	40.4	9.96
m3d diffusivity	36.87	23.19	30.02	0.0E+00	6.8E+00	364.5	12.68	297.6	42.90	250.8	19.09	239.3	15.02	198.8	31.92	203.0	18.47
m3d viscosity	50.07	24.86	28.59	8.6E-15	2.9E+00	364.5	12.62	314.0	46.09	249.4	18.95	246.1	14.68	209.2	32.66	207.5	19.45
h3d temp	65.70	23.54	31.56	-7.7E+01	1.0E+35	95.4	3.77	75.8	14.56	59.3	4.64	53.0	4.27	44.1	8.04	44.1	5.06
h3d pressure	81.82	24.13	31.58	-3.4E+03	1.0E+35	95.4	3.78	82.3	12.00	64.3	5.14	52.9	4.87	45.0	7.78	45.2	5.34
h3d x velocity	84.18	24.18	31.55	-5.3E+01	1.0E+35	95.4	3.89	86.1	11.27	67.4	6.22	63.3	4.59	54.5	8.86	53.8	5.53
h3d y velocity	84.32	24.18	31.55	-4.6E+01	1.0E+35	95.4	3.83	84.5	11.42	67.1	5.74	62.3	5.04	53.5	8.64	53.8	5.53
h3d z velocity	86.82	24.24	31.54	-3.2E+00	1.0E+35	95.4	3.87	88.4	10.76	85.6	8.50	76.9	5.29	68.9	9.83	69.1	6.65
M3d density	40.14	18.81	22.59	1.0E+00	3.0E+00	288.0	11.28	136.8	41.91	160.3	11.63	121.6	10.94	-	-	105.2	11.63
M3d pressure	100.00	25.17	63.00	-2.2E+00	2.2E+00	288.0	11.20	272.6	35.18	237.3	14.91	225.1	16.59	-	-	208.4	17.20
M3d x velocity	100.00	25.17	63.00	-2.2E+00	2.3E+00	288.0	10.83	275.6	32.30	230.4	14.73	215.1	15.91	-	-	197.7	16.84
M3d y velocity	100.00	25.17	63.00	-2.1E+00	2.3E+00	288.0	10.54	275.1	32.19	223.1	14.27	215.2	15.16	-	-	197.7	16.65
M3d z velocity	100.00	25.17	63.00	-5.2E+00	9.0E+00	288.0	10.32	275.5	32.62	226.6	14.74	213.7	16.05	-	-	196.8	16.14
atom x position	61.10	23.82	31.01	-4.8E-02	4.6E+02	107.7	7.07	84.3	21.18	76.0	7.88	78.8	7.61	67.3	12.88	68.6	9.07
atom y position	45.90	23.32	26.99	3.7E-02	2.1E+03	107.7	7.08	65.9	30.76	60.4	6.97	56.4	6.31	47.0	10.49	46.9	7.73
atom z position	61.68	23.84	27.48	9.1E-05	4.6E+02	107.7	7.46	94.6	19.86	82.6	9.00	86.1	8.25	75.7	13.80	78.2	9.93
atom y velocity	64.65	23.87	30.96	1.5E-01	1.4E-01	107.7	7.30	95.7	19.88	93.8	10.07	99.1	9.65	84.8	14.93	87.6	9.92
atom temp	64.91	23.94	27.41	3.0E-03	7.1E+03	107.7	6.69	95.7	19.76	91.6	10.27	95.9	8.34	84.6	15.02	84.6	10.31
atom energy	3.45	18.57	21.79	-3.6E+00	-2.7E+00	107.7	7.15	77.9	38.59	74.1	7.98	71.8	7.01	60.8	12.66	60.5	8.30
lucy	61.39	24.38	31.09	-6.1E-02	1.2E+03	160.5	-	137.8	-	99.5	-	90.0	-	73.6	-	77.8	-
dauid10m	25.23	17.08	31.11	-4.4E+03	1.8E+03	322.5	-	144.9	-	155.7	-	163.4	-	108.6	-	131.9	-
torso	84.72	18.48	31.08	-2.7E+02	5.8E+02	1.9	-	1.7	-	1.5	-	1.5	-	1.3	-	1.3	-
rhl	71.90	20.14	25.99	1.5E+00	3.6E+02	8.4	-	7.1	-	5.8	-	5.6	-	4.7	-	4.8	-

Table 1. Compression results for the Miranda (m2d, m3d, M3d) structured grids, the atom point set, the lucy and david triangle meshes, and the torso and rhl tetrahedral meshes. All data but M3d is represented in single precision. The [ILS2005] scheme operates on single precision only, hence the missing values. For the meshes we report only the compressed size of vertex coordinates; timings are dominated by connectivity coding, and are hence excluded. The range measures the (logarithm of) the number of floating-point values between min and max. Note that the first-order entropy is limited by the number of samples in a data set.

tation.) Arguably such data sets should use an integer rather than floating-point representation, although for simplicity or other reasons it is common practice to use floating-point. Contrary to [16], which entropy codes all bits of the residual, our new coder sacrifices such potential compression gains for speed by storing these repeated low-order bits in raw and uncompressed form. However, the massive data sets from scientific simulation that motivated our work on high-speed compression, as well as our tetrahedral meshes, rarely exhibit any significant low-order redundancy, as also evidenced by our results.

### 5.1.1 Lossy Compression

Fig. 3 shows that our scheme gracefully adapts to decreasing levels of precision when discarding the least significant mantissa (and eventually exponent) bits. For  $n$  bits of precision, the schemes [7, 22] require  $\log_2 n$  bits to code the number of leading zeros, whereas our scheme exploits the combination of low entropy in the leading-zero count and the elimination of the low-order bits that are most difficult to predict and compress.

### 5.2 Compression Speed

Fig. 4 shows the speed of compressing from memory to disk, including disk write time. Because the throughput of our method, its decompression speed is similar to its compression speed.) We also include the raw I/O performance of dumping the data uncompressed using a single `fwrite` call. Timings

correspond to the median of five runs. Whereas our compressor is slightly slower than the less effective compressors [7, 22], it is nearly twice as fast as [16] while producing similar compression rates. However, in more I/O-intensive scenarios, such as in massively parallel simulations dumping data to the same file system (as is common), the improved compression of our method over [7, 22] results in a net gain in effective throughput. We integrated our compression code with Miranda's dump routines, and ran performance tests on 276 nodes of LLNL's M3d supercomputer. Achieving on average a lossless reduction of 3.7 on 75 GB of data dumped, the overall dump time was reduced by a factor of 2.7 over writing the data uncompressed.

### 5.3 Entropy Coding

We compared the raw throughput of our range coder and Schindler's [23] by (1) passing raw bytes through it with no compression and (2) entropy coding byte sequences. In both cases, the source data was the uncompressed floating-point data used in our experiments. Timings show that our coder is 40% faster for raw transmission and 28% faster for entropy coding. Meanwhile, the inefficiency of our coder due to loss of precision and range reduction is only 26 bytes of overhead for 1.5 GB of coded data. Its raw throughput is only 20% less than an `fwrite` call, while its entropy coding throughput of 20 MB per second, which includes probability modeling and I/O time, compares favorably with state-of-the-art entropy coders [25].

Lindström et al. [2006]

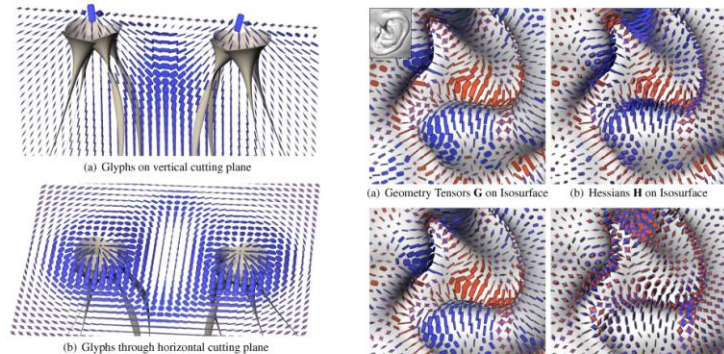


Fig. 8. Glyphs in the double point load stress tensor field reveal the minor eigenvector along which hyperstreamlines [7] are traced (a), and the variation in stress with distance from the load (b).

volume rendered [32], but its eigenvectors are commonly used in non-photo-realistic rendering, e.g. curvature-based strokes [11, 14, 19]. Inspecting geometry tensors could help debug an NPR method giving unexpected results in an unfamiliar dataset. Fig. 9(a) visualizes geometry tensors  $G$  on an isosurface (sampled by a particle system [39]) of an ear from the Visible Human male CT scan. Variations in surface curvature are reflected in the new glyphs: convex (blue circles), concave (orange circles), and saddles (orange and blue stars). For comparison, Fig. 9(b) shows the full Hessian  $H$  from which  $G$  was computed.

The new glyphs may also have a role in visualizing the tensor ingredients of image analysis methods such as edge detection. One edge definition is zero-crossing on the second directional derivative along the gradient direction,  $f'' = n^T H n$ . This edge surface is sampled by a particle system [33] in Fig. 9(c), showing the Hessians at the edge locations, and revealing close similarities with the geometry tensors on the isosurface in Fig. 9(a), indicating that one of the Hessian eigenvalues is near zero even though this is not part of the edge definition. Another edge definition is the zero-crossing of the Laplacian  $\nabla^2 f = \text{tr}(H)$ , and Fig. 9(d) illustrates the difference between the Hessians on this surface and those in Fig. 9(c). The consistently gray glyph halos in Fig. 9(d) indicate that these are traceless tensors.

As a demonstration of the glyphs in a 2-D visualization, Fig. 10 visualizes a cross-section of a simulation of jet flow rightward into a steady medium, causing turbulence. Glyphs of rate-of-deformation tensors document how an infinitesimal volume is stretched or compressed as it moves along the flow. A backdrop of line integral convolution [4] (with contrast modulated by velocity) provides visual context. Fig. 10(a) uses the exponentially-scaled ellipses of [34] to map tensors with negative eigenvalues to positive-definite tensors suitable for ellipsoid visualization. When the absolute difference between eigenvalues becomes too large, these glyphs can become so stretched that they overlap each other and extend over a significant portion of the domain, undermining the locality normally enjoyed by glyphs. Such stretching also reduces the visual presence of the needle-like glyphs for tensors with larger norms, contrary to scale preservation (6). Fig. 10(b) uses our superquadratic glyphs with  $s(\|D\|) \approx \|D\|$ . The aspect ratio reflects the relative eigenvalue magnitudes, the size correctly indicates the tensor norm, and pointed glyph shapes clearly communicate eigenvector directions. With compression of scale variation

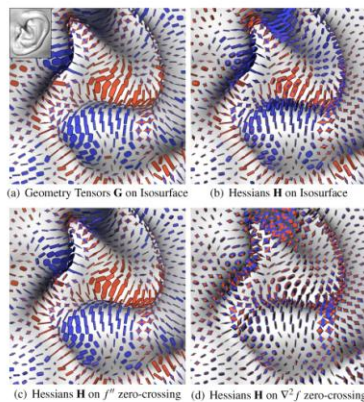


Fig. 9. Visualization of geometry tensors (a) and Hessians associated with isosurfaces (b) and two different definitions of edges, zero-crossings of the second-directional derivative (c) and the Laplacian (d). These results use  $s(\|D\|) \approx \|D\|^{1/2}$  in (6).

$s(\|D\|) \approx \|D\|^{1/2}$ , Fig. 10(c) better shows the directional patterns where the tensor norm is low. Colormapping the rate-of-deformation tensor trace with glyph halos highlights the regions of over-all stretching or compression, especially along the bottom edge of the domain.

Finally, Fig. 11 demonstrates how our new glyph performs traceless tensor visualization, in a side-by-side comparison to the dedicated traceless NLCT tensor glyphs by Jankun-Kelly et al. [25]. Traceless tensors form a plane in eigenvalue space, and we are visualizing samples from a square within this plane, centered around the zero tensor (cf. Fig. 4(c)). Unlike the traceless glyph, which maps tensor norm to glyph sharpness, our glyph expresses norm by its overall scale  $s(\|D\|) \approx \|D\|$ . Consequently, the traceless glyph requires pre-specification of maximum eigenvalue magnitudes (which are mapped to perfect sharpness), while our glyph can be used without such prior information. Another notable difference is that limiting their glyph to traceless tensors allows Jankun-Kelly et al. to make use of parts of the superquadratic shape space – including cylinders and boxes – that our approach sets aside for positive- or negative-definite tensors.

### 6 CONCLUSION

Visualization research has made significant progress in visualizing second-order tensor fields, but has mostly concentrated on the positive-definite case. Faced with indefinite tensors, a frequent strategy is to map them to positive-definite tensors prior to visualization [34, 22, 21, 52, 33]. Even when bijective mappings are used (so mathematically, no information is lost), such mappings still visually obscure the difference between positive and negative eigenvalues, which is a fundamental qualitative aspect in various applications.

Therefore, we propose an extension of a previous positive-definite tensor glyph [28] to the full space of symmetric second-order tensors. Our glyph emphasizes differences in eigenvalue sign in a way that, unlike the Reynolds glyph [18], prevents small eigenvalues from being occluded by larger ones. We also propose to use halos to ensure correct indication of the tensor norm, and pointed glyph shapes clearly communicate eigenvector directions. With compression of scale variation

Schultz & Kindlmann [2010]

two cases was not a factor that could affect the outcome of this study negatively. One experimenter engaged in the demonstration of the table and in assisting the participant when needed. A second experimenter took notes and also documented the session by voice recording. A prepared interview guide was used. It included a set of predefined questions that covered various aspects of the design goals (section 5.1) and also a number of potential questions used to prompt the participant to “think aloud” when needed. Both experimenters engaged in the conversation and made sure that all questions in the interview guide were covered by the end of the session. Some questions were discussed while participants worked on the cases and some were reviewed afterwards. After using the table participants completed a subjective satisfaction questionnaire. The responses were given on a 5-point rating scale: *Strongly unfavorable* (1), *Unfavorable* (2), *Unsure* (3), *Favorable* (4), and *Strongly favorable* (5). The questionnaire covered the following

- Overall impression:** The overall impression of the table.
- Willingness to use:** Whether the orthopedic surgeon would like to use the table in their daily work.
- Efficiency for planning:** Whether using the table would save time for pre-operative planning.
- Efficiency for surgery:** Whether using the table pre-operatively would save time during actual surgery.
- Quality of work:** Whether using the table would improve the quality of clinical work.
- Learnability:** Ease of learning table usage for a novice user.
- Interaction:** Ease of interacting with the table.
- Benefits over static imaging:** Whether interactive 3D imaging is superior to the series of static 3D snap-shots used today.
- Collaboration:** Whether access to the table at work would facilitate collaboration between several people.
- Similarity to real situation:** Whether the similarity to a real situation (patient lying on a table) facilitates insights and decision making.
- Recommend to others:** Whether the orthopedic surgeon would recommend colleagues to use the table at work.

### 7 RESULTS

The user study proved effective for the objective of collecting distinct and broad feedback from the orthopedic surgeons about how the visualization table would fit in their application domain. They did not consider the “think aloud” approach to be distracting from the evaluation tasks. The overall assessment from the surgeons is that the table would be useful in their clinical work. This is illustrated by the numerical ratings in the post-session questionnaire, see figure 12. Responses for the eleven statements has a group mean value<sup>1</sup> of 3.8 (two statements), 4 (two statements) and above 4 (seven statements) respectively, all corresponding to a clearly favorable rating. Statistical significance was, however, not achieved but this is to be expected for this small study. There is only one example of a negative rating, one surgeon expressed moderate disagreement with the table’s potential to improve efficiency during surgery (specialist, age 50). There were three statements concerning general impressions of usefulness, *Overall impression*, *Willingness to use*, and *Recommend to others*, and in all three cases the study shows a strongly favorable mean rating of 4.4. Both the younger and less experienced participants and the older specialists contributed

<sup>1</sup>It can be discussed whether averaging in an ordinal scale is appropriate, our conclusion is that in this case it is the best way to convey the results, in combination with the min-max measures in figure 12.

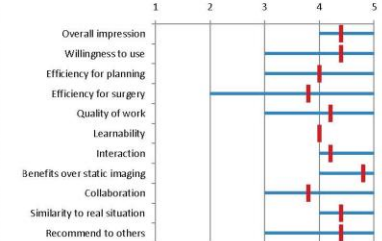


Fig. 12. The quantitative results of the user study questionnaire. Subjective satisfaction regarding use of the table was measured for 11 questions, see section 6. The 5-point rating scale ranges from *Strongly unfavorable* (1) through *Unsure* (3) to *Strongly favorable* (5). Vertical red bars denote the mean value and horizontal blue lines denote the full span of given ratings.

to all levels of the rating scale (3-5). Hence, for this participant pool, age and level of experience did not seem to affect the attitude towards the technique.

The issues behind the rather general statements in the questionnaire were discussed in greater detail during the sessions at the table and these findings provide a more nuanced and informative view of opinions. Below, these findings are summarized under the following four subheadings: Ease of use and learnability, Clinical usefulness, Workflow, and Desired features.

**7.1 Ease of use and learnability**  
Low learning threshold and high usability were central objectives in the design of the system, reflected by design requirements R1-R5. Regarding the overall impression of the table all participants expressed positive statements. The interface was considered intuitive and convenient, and it was easy to learn how to use the basic functionality. The comments about learning threshold expressed an anticipation that novice users would quickly learn the basic functionality, although some of the more advanced functionality (activated via the pucks) would require some practice. All appreciated the clean interface with only a few visible GUI elements and emphasized the benefit and importance of the screen being focused towards visualizing the 3D image. Regarding the interaction, the touch gestures were described as intuitive and straightforward to use, also for one of the participants who pointed out that he had never used a touch-controlled interface before (specialist, age 54). The surgeons were asked if they perceived the interaction as robust and responsive. They all concurred, through statements that the result of actions on the screen was what they expected and that they felt in control. Nobody mentioned that the level of precision provided by the touch technology and the RST interaction was insufficient or problematic. Even though the participants did not bring it up, the experimenters noted a few occasions of unintentional gestures due to holding the knuckles of inactive fingers too close to the surface. The typical effect was that panning occurred instead of an intended x-y-rotation, which the users dealt with by lifting the hand and reapplying the rotation gesture. For the additional MPR slice views it was commented that touch gestures were more efficient for transversal browsing than using a mouse. The pucks were, in general, described as a convenient approach for reaching additional features. The interaction that caused some confusion was the advanced parts of the clip plane functionality, namely to control and understand slab clipping. The surgeons adopted the natural size zoom as an integral part of the toolset and no usability obstacles

Lundström et al. [2011]

# Results: evaluation scenarios (selection)

algorithmic performance  
35% of scenarios

qualitative result inspection  
46% of scenarios

user performance/experience  
14% of scenarios

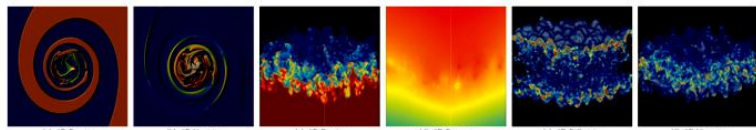


Fig. 1. Visualizations of 2D data (as pseudocolored height fields) and 3D data (volume rendered) used in our experiments.

name	data set				compressed size (MB) and compression time (seconds)											
	unique entropy (%)	range (bits)	min	max	size (MB)	time (sec)	zlib	[RKB2006]	[EFF2000]	[ILS2005]	new scheme	zlib	[RKB2006]	[EFF2000]	[ILS2005]	new scheme
m2d density	3.89	3.49 21.83	8.7E-01	1.2E+00	19.6	0.71	1.6	0.86	4.3	0.49	4.4	0.56	1.3	1.08	1.3	0.56
m2d vorticity	99.20	22.25 31.05	-1.4E+02	2.5E+01	19.6	0.71	18.4	2.14	11.8	1.21	15.5	1.29	12.9	2.22	13.8	1.49
m3d density	7.67	5.16 23.60	1.0E+00	3.0E+00	364.5	12.81	50.4	17.55	100.5	9.06	96.3	8.48	35.7	19.03	35.5	9.25
m3d pressure	27.29	23.91 31.06	-3.7E+00	2.3E+03	364.5	12.80	229.2	99.76	95.6	9.31	87.9	8.87	40.1	18.79	40.4	9.96
m3d diffusivity	36.87	23.19 30.02	0.0E+00	6.8E+00	364.5	12.68	297.6	42.90	250.8	19.09	239.3	15.02	198.8	31.92	203.0	18.47
m3d viscosity	50.07	24.86 28.59	8.6E-15	2.9E+00	364.5	12.62	314.0	46.09	249.4	18.95	246.1	14.68	209.2	32.66	207.5	19.45
h3d temp	65.70	23.54 31.56	-7.7E+01	1.0E+35	95.4	3.77	75.8	14.56	59.3	4.64	53.0	4.27	44.1	8.04	44.1	5.06
h3d pressure	81.82	24.13 31.58	-3.4E+03	1.0E+35	95.4	3.78	82.3	12.00	64.3	5.14	52.9	4.87	45.0	7.78	45.2	5.34
h3d x velocity	84.18	24.18 31.55	-5.3E+01	1.0E+35	95.4	3.89	86.1	11.27	67.4	6.22	63.3	4.59	54.5	8.86	53.4	5.44
h3d y velocity	84.32	24.18 31.55	-4.6E+01	1.0E+35	95.4	3.83	84.5	11.42	67.1	5.74	62.3	5.04	53.5	8.64	53.8	5.53
h3d z velocity	86.82	24.24 31.54	-3.2E+00	1.0E+35	95.4	3.87	88.4	10.76	85.6	8.50	76.9	5.29	68.9	9.83	69.1	6.65
M3d density	40.14	18.81 22.59	1.0E+00	3.0E+00	288.0	11.28	136.8	41.91	160.3	11.63	121.6	10.94	-	-	105.2	11.63
M3d pressure	100.00	25.17 63.00	-2.2E+00	2.2E+00	288.0	11.20	272.6	35.18	237.3	14.91	225.1	16.59	-	-	208.4	17.20
M3d x velocity	100.00	25.17 63.00	-2.2E+00	2.3E+00	288.0	10.83	275.6	32.30	230.4	14.73	215.1	15.91	-	-	197.7	16.84
M3d y velocity	100.00	25.17 63.00	-2.1E+00	2.3E+00	288.0	10.54	275.1	32.19	223.1	14.27	215.2	15.16	-	-	197.7	16.65
M3d z velocity	100.00	25.17 63.00	-5.2E+00	9.0E+00	288.0	10.32	275.5	32.62	226.6	14.74	213.7	16.05	-	-	196.8	16.14
atom x position	61.10	23.82 31.01	-4.8E-02	4.6E+02	107.7	7.07	84.3	21.18	76.0	7.88	78.8	7.61	67.3	12.88	68.6	9.07
atom y position	45.90	23.32 26.99	3.7E-02	2.1E+03	107.7	7.08	65.9	30.76	60.4	6.97	56.4	6.31	47.0	10.49	46.9	7.73
atom z position	61.68	23.84 27.48	9.1E-05	4.6E+02	107.7	7.46	94.6	19.86	82.6	9.00	86.1	8.25	75.7	13.80	78.2	9.93
atom y velocity	64.65	23.87 30.96	1.5E-01	1.4E-01	107.7	7.30	95.7	19.88	93.8	10.07	99.1	9.65	84.8	14.93	87.6	9.92
atom temp	64.91	23.94 27.41	3.0E-03	7.1E+03	107.7	6.69	95.7	19.76	91.6	10.27	95.9	8.34	84.6	15.02	84.6	10.31
atom energy	3.45	18.57 21.79	-3.6E+00	-2.7E+00	107.7	7.15	77.9	38.59	74.1	7.98	71.8	7.01	60.8	12.66	60.5	8.30
lucy	61.39	24.38 31.09	-6.1E-02	1.2E+03	160.5	-	137.8	-	99.5	-	90.0	-	73.6	-	77.8	-
dauid10mm	25.23	17.08 31.11	-4.4E+03	1.8E+03	322.5	-	144.9	-	155.7	-	163.4	-	108.6	-	131.9	-
torso	84.72	18.48 31.08	-2.7E+02	5.8E+02	1.9	-	1.7	-	1.5	-	1.5	-	1.3	-	1.3	-
rhl	71.90	20.14 25.99	1.5E+00	3.6E+02	8.4	-	7.1	-	5.8	-	5.6	-	4.7	-	4.8	-

Table 1. Compression results for the Miranda (m2d, m3d, M3d) structured grids, the atom point set, the lucy and dauid triangle meshes, and the torso and rhl tetrahedral meshes. All data but M3d is represented in single precision. The [ILS2005] scheme operates on single precision only, hence the missing values. For the meshes we report only the compressed size of vertex coordinates; timings are dominated by connectivity coding, and are hence excluded. The range measures the (logarithm of) the number of floating-point values between min and max. Note that the first-order entropy is limited by the number of samples in a data set.

tation.) Arguably such data sets should use an integer rather than floating-point representation, although for simplicity or other reasons it is common practice to use floating-point. Contrary to [16], which entropy codes all bits of the residual, our new coder sacrifices such potential compression gains for speed by storing these repeated low-order bits in raw and uncompressed form. However, the massive data sets from scientific simulation that motivated our work on high-speed compression, as well as our tetrahedral meshes, rarely exhibit any significant low-order redundancy, as also evidenced by our results.

### 5.1.1 Lossy Compression

Fig. 3 shows that our scheme gracefully adapts to decreasing levels of precision when discarding the least significant mantissa (and eventually exponent) bits. For  $n$  bits of precision, the schemes [7, 22] require  $\log_2 n$  bits to code the number of leading zeros, whereas our scheme exploits the combination of low entropy in the leading-zero count and the elimination of the low-order bits that are most difficult to predict and compress.

### 5.2 Compression Speed

Fig. 4 shows the speed of compressing from memory to disk, including disk write time. Because the throughput of our method, its decompression speed is similar to its compression speed.) We also include the raw I/O performance of dumping the data uncompressed using a single `fwrite` call. Timings

correspond to the median of five runs. Whereas our compressor is slightly slower than the less effective compressors [7, 22], it is nearly twice as fast as [16] while producing similar compression rates. However, in more I/O-intensive scenarios, such as in massively parallel simulations dumping data to the same file system (as is common), the improved compression of our method over [7, 22] results in a net gain in effective throughput. We integrated our compression code with Miranda's dump routines, and ran performance tests on 276 nodes of LLNL's MCR supercomputer. Achieving on average a lossless reduction of 3.7 on 75 GB of data dumped, the overall dump time was reduced by a factor of 2.7 over writing the data uncompressed.

### 5.3 Entropy Coding

We compared the raw throughput of our range coder and Schindler's [23] by (1) passing raw bytes through it with no compression and (2) entropy coding byte sequences. In both cases, the source data was the uncompressed floating-point data used in our experiments. Timings show that our coder is 40% faster for raw transmission and 28% faster for entropy coding. Meanwhile, the inefficiency of our coder due to loss of precision and range reduction is only 26 bytes of overhead for 1.5 GB of coded data. Its raw throughput is only 20% less than an `fwrite` call, while its entropy coding throughput of 20 MB per second, which includes probability modeling and I/O time, compares favorably with state-of-the-art entropy coders [25].

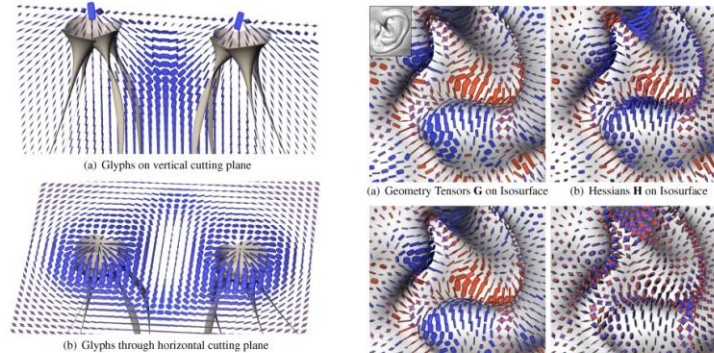


Fig. 8. Glyphs in the double point load stress tensor field reveal the minor eigenvector along which hyperstreamlines [7] are traced (a), and the variation in stress with distance from the load (b).

volume rendered [32], but its eigenvectors are commonly used in non-photo-realistic rendering, e.g. curvature-based strokes [11, 14, 19]. Inspecting geometry tensors could help debug an NPR method giving unexpected results in an unfamiliar dataset. Fig. 9(a) visualizes geometry tensors  $G$  on an isosurface (sampled by a particle system [39]) of an ear from the Visible Human male CT scan. Variations in surface curvature are reflected in the new glyphs: convex (blue circles), concave (orange circles), and saddles (orange and blue stars). For comparison, Fig. 9(b) shows the full Hessian  $H$  from which  $G$  was computed.

The new glyphs may also have a role in visualizing the tensor ingredients of image analysis methods such as edge detection. One edge definition is zero-crossing on the second directional derivative along the gradient direction,  $f'' = \mathbf{n}^T H \mathbf{n}$ . This edge surface is sampled by a particle system [33] in Fig. 9(c), showing the Hessians at the edge locations, and revealing close similarities with the geometry tensors on the isosurface in Fig. 9(a), indicating that one of the Hessian eigenvalues is near zero even though this is not part of the edge definition. Another edge definition is the zero-crossing of the Laplacian  $\nabla^2 f = \text{tr}(H)$ , and Fig. 9(d) illustrates the difference between the Hessians on this surface and those in Fig. 9(c). The consistently gray glyph halos in Fig. 9(d) indicate that these are traceless tensors.

As a demonstration of the glyphs in a 2-D visualization, Fig. 10 visualizes a cross-section of a simulation of jet flow rightward into a steady medium, causing turbulence. Glyphs of rate-of-deformations tensors document how an infinitesimal volume is stretched or compressed as it moves along the flow. A backdrop of line integral convolution [4] (with contrast modulated by velocity) provides visual context. Fig. 10(a) uses the exponentially-scaled ellipses of [34] to map tensors with negative eigenvalues to positive-definite tensors suitable for ellipsoid visualization. When the absolute difference between eigenvalues becomes too large, these glyphs can become so stretched that they overlap each other and extend over a significant portion of the domain, undermining the locality normally enjoyed by glyphs. Such stretching also reduces the visual presence of the needle-like glyphs for tensors with larger norms, contrary to scale preservation (6). Fig. 10(b) uses our superquadratic glyphs with  $s(\|D\|) \approx \|D\|$ . The aspect ratio reflects the relative eigenvalue magnitudes, the size correctly indicates the tensor norm, and pointed glyph shapes clearly communicate eigenvector directions. With compression of scale variation

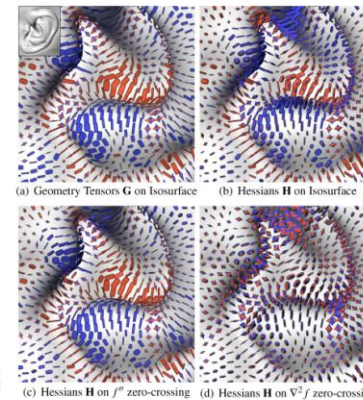


Fig. 9. Visualization of geometry tensors (a) and Hessians associated with isosurfaces (b) and two different definitions of edges, zero-crossings of the second-directional derivative (c) and the Laplacian (d). These results use  $s(\|D\|) \approx \|D\|^{1/2}$  in (6).

$s(\|D\|) \approx \|D\|^{1/2}$ , Fig. 10(c) better shows the directional patterns where the tensor norm is low. Colormapping the rate-of-deformation tensor trace with glyph halos highlights the regions of over-all stretching or compression, especially along the bottom edge of the domain.

Finally, Fig. 11 demonstrates how our new glyph performs traceless tensor visualization, in a side-by-side comparison to the dedicated traceless NLCT tensor glyphs by Jankun-Kelly et al. [25]. Traceless tensors form a plane in eigenvalue space, and we are visualizing samples from a square within this plane, centered around the zero tensor (cf. Fig. 4(c)). Unlike the traceless glyph, which maps tensor norm to glyph sharpness, our glyph expresses norm by its overall scale  $s(\|D\|) \approx \|D\|$ . Consequently, the traceless glyph requires pre-specification of maximum eigenvalue magnitudes (which are mapped to perfect sharpness), while our glyph can be used without such prior information. Another notable difference is that limiting their glyph to traceless tensors allows Jankun-Kelly et al. to make use of parts of the superquadratic shape space – including cylinders and boxes – that our approach sets aside for positive- or negative-definite tensors.

### 6 CONCLUSION

Visualization research has made significant progress in visualizing second-order tensor fields, but has mostly concentrated on the positive-definite case. Faced with indefinite tensors, a frequent strategy is to map them to positive-definite tensors prior to visualization [34, 22, 21, 52, 33]. Even when bijective mappings are used (so mathematically, no information is lost), such mappings still visually obscure the difference between positive and negative eigenvalues, which is a fundamental qualitative aspect in various applications.

Therefore, we propose an extension of a previous positive-definite tensor glyph [28] to the full space of symmetric second-order tensors. Our glyph emphasizes differences in eigenvalue sign in a way that, unlike the Reynolds glyph [18], prevents small eigenvalues from being occluded by larger ones. We also propose to use halos to ensure tensor glyph visibility even when one or more eigenvalues are near zero. Finally, we present a time- and memory-efficient implementa-

two cases was not a factor that could affect the outcome of this study negatively. One experimenter engaged in the demonstration of the table and in assisting the participant when needed. A second experimenter took notes and also documented the session by voice recording. A prepared interview guide was used. It included a set of predefined questions that covered various aspects of the design goals (section 5.1) and also a number of potential questions used to prompt the participant to “think aloud” when needed. Both experimenters engaged in the conversation and made sure that all questions in the interview guide were covered by the end of the session. Some questions were discussed while participants worked on the cases and some were reviewed afterwards.

After using the table participants completed a subjective satisfaction questionnaire. The responses were given on a 5-point rating scale: *Strongly unfavorable* (1), *Unfavorable* (2), *Unsure* (3), *Favorable* (4), and *Strongly favorable* (5). The questionnaire covered the following

1. **Overall impression:** The overall impression of the table.
2. **Willingness to use:** Whether the orthopedic surgeon would like to use the table in their daily work.
3. **Efficiency for planning:** Whether using the table would save time for pre-operative planning.
4. **Efficiency for surgery:** Whether using the table pre-operatively would save time during actual surgery.
5. **Quality of work:** Whether using the table would improve the quality of clinical work.
6. **Learnability:** Ease of learning table usage for a novice user.
7. **Interaction:** Ease of interacting with the table.
8. **Benefits over static imaging:** Whether interactive 3D imaging is superior to the series of static 3D snap-shots used today.
9. **Collaboration:** Whether access to the table at work would facilitate collaboration between several people.
10. **Similarity to real situation:** Whether the similarity to a real situation (patient lying on a table) facilitates insights and decision making.
11. **Recommend to others:** Whether the orthopedic surgeon would recommend colleagues to use the table at work.

### 7 RESULTS

The user study proved effective for the objective of collecting distinct and broad feedback from the orthopedic surgeons about how the visualization table would fit in their application domain. They did not consider the “think aloud” approach to be distracting from the evaluation tasks. The overall assessment from the surgeons is that the table would be useful in their clinical work. This is illustrated by the numerical ratings in the post-session questionnaire, see figure 12. Responses for the eleven statements has a group mean value<sup>1</sup> of 3.8 (two statements), 4 (two statements) and above 4 (seven statements) respectively, all corresponding to a clearly favorable rating. Statistical significance was, however, not achieved but this is to be expected for this small study. There is only one example of a negative rating, one surgeon expressed moderate disagreement with the table’s potential to improve efficiency during surgery (specialist, age 50). There were three statements concerning general impressions of usefulness, *Overall impression*, *Willingness to use*, and *Recommend to others*, and in all three cases the study shows a strongly favorable mean rating of 4.4. Both the younger and less experienced participants and the older specialists contributed

<sup>1</sup>It can be discussed whether averaging in an ordinal scale is appropriate, our conclusion is that in this case it is the best way to convey the results, in combination with the min-max measures in figure 12.

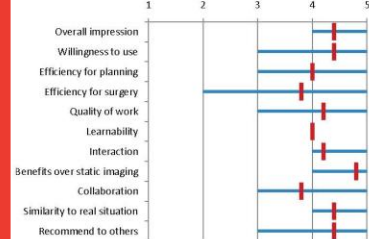


Fig. 12. The quantitative results of the user study questionnaire. Subjective satisfaction regarding use of the table was measured for 11 questions, see section 6. The 5-point rating scale ranges from *Strongly unfavorable* (1) through *Unsure* (3) to *Strongly favorable* (5). Vertical red bars denote the mean value and horizontal blue lines denote the full span of given ratings.

age and level of experience did not seem to affect the attitude towards the technique.

The issues behind the rather general statements in the questionnaire were discussed in greater detail during the sessions at the table and these findings provide a more nuanced and informative view of opinions. Below, these findings are summarized under the following four subheadings: Ease of use and learnability, Clinical usefulness, Workflow, and Desired features.

### 7.1 Ease of use and learnability

Low learning threshold and high usability were central objectives in the design of the system, reflected by design requirements R1-R5. Regarding the overall impression of the table all participants expressed positive statements. The interface was considered intuitive and convenient, and it was easy to learn how to use the basic functionality. The comments about learning threshold expressed an anticipation that novice users would quickly learn the basic functionality, although some of the more advanced functionality (activated via the pucks) would require some practice. All appreciated the clean interface with only a few visible GUI elements and emphasized the benefit and importance of the screen being focused towards visualizing the 3D image.

Regarding the interaction, the touch gestures were described as intuitive and straightforward to use, also for one of the participants who pointed out that he had never used a touch-controlled interface before (specialist, age 54). The surgeons were asked if they perceived the interaction as robust and responsive. They all concurred, through statements that the result of actions on the screen was what they expected and that they felt in control. Nobody mentioned that the level of precision provided by the touch technology and the RST interaction was insufficient or problematic. Even though the participants did not bring it up, the experimenters noted a few occasions of unintentional gestures due to holding the knuckles of inactive fingers too close to the surface. The typical effect was that panning occurred instead of an intended x-y-rotation, which the users dealt with by lifting the hand and reapplying the rotation gesture. For the additional MPR slice views it was commented that touch gestures were more efficient for transversal browsing than using a mouse.

The pucks were, in general, described as a convenient approach for reaching additional features. The interaction that caused some confusion was the advanced parts of the clip plane functionality, namely to control and understand slab clipping. The surgeons adopted the natural size zoom as an integral part of the toolset and no usability obstacles

# Results: evaluation scenarios (selection)

algorithmic performance  
35% of scenarios

qualitative result inspection  
46% of scenarios

user performance/experience  
14% of scenarios

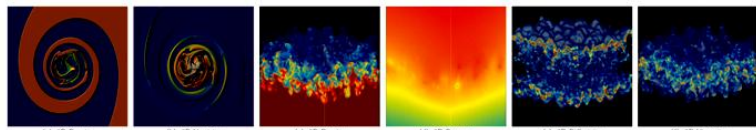


Fig. 1. Visualizations of 2D data (as pseudocolored height fields) and 3D data (volume rendered) used in our experiments.

name	data set					compressed size (MB) and compression time (seconds)											
	unique entropy (%)	entropy (bits)	range (bits)	min	max	size (MB)	time (sec)	zlib	[RKB2006]	[EFF2000]	[ILS2005]	new scheme					
m2d density	3.89	3.49	21.83	8.7E-01	1.2E+00	19.6	0.71	1.6	0.86	4.3	0.49	4.4	0.56	1.3	1.08	1.3	0.56
m2d vorticity	99.20	22.25	31.05	-1.4E+02	2.5E+01	19.6	0.71	18.4	2.14	11.8	1.21	15.5	1.29	12.9	2.22	13.8	1.49
m3d density	7.67	5.16	23.60	1.0E+00	3.0E+00	364.5	12.81	50.4	17.55	100.5	9.06	96.3	8.48	35.7	19.03	35.5	9.25
m3d pressure	27.29	23.91	31.06	-3.7E+00	2.3E+03	364.5	12.80	229.2	99.76	95.6	9.31	87.9	8.87	40.1	18.79	40.4	9.96
m3d diffusivity	36.87	23.19	30.02	0.0E+00	6.8E+00	364.5	12.68	297.6	42.90	250.8	19.00	239.3	15.02	198.8	31.92	203.0	18.47
m3d viscosity	50.07	24.86	28.59	8.6E-15	2.9E+00	364.5	12.62	314.0	46.09	249.4	18.95	246.1	14.68	209.2	32.66	207.5	19.45
h3d temp	65.70	23.54	31.56	-7.7E+01	1.0E+35	95.4	3.77	75.8	14.56	59.3	4.64	53.0	4.27	44.1	8.04	44.1	5.06
h3d pressure	81.82	24.13	31.58	-3.4E+03	1.0E+35	95.4	3.78	82.3	12.00	64.3	5.14	52.9	4.87	45.0	7.78	45.2	5.34
h3d x velocity	84.18	24.18	31.55	-5.3E+01	1.0E+35	95.4	3.89	86.1	11.27	67.4	6.22	63.3	4.59	54.5	8.86	53.8	5.53
h3d y velocity	84.32	24.18	31.55	-4.6E+01	1.0E+35	95.4	3.83	84.5	11.42	67.1	5.74	62.3	5.04	53.5	8.64	53.8	5.53
h3d z velocity	86.82	24.24	31.54	-3.2E+00	1.0E+35	95.4	3.87	88.4	10.76	85.6	8.50	76.9	5.29	68.9	9.83	69.1	6.65
M3d density	40.14	18.81	22.59	1.0E+00	3.0E+00	288.0	11.28	136.8	41.91	160.3	11.63	121.6	10.94	-	-	105.2	11.63
M3d pressure	100.00	25.17	63.00	-2.2E+00	2.2E+00	288.0	11.20	272.6	35.18	237.3	14.91	225.1	16.59	-	-	208.4	17.20
M3d x velocity	100.00	25.17	63.00	-2.2E+00	2.3E+00	288.0	10.83	275.6	32.30	230.4	14.73	215.1	15.91	-	-	197.7	16.84
M3d y velocity	100.00	25.17	63.00	-2.1E+00	2.3E+00	288.0	10.54	275.1	32.19	223.1	14.27	215.2	15.16	-	-	197.7	16.65
M3d z velocity	100.00	25.17	63.00	-5.2E+00	9.0E+00	288.0	10.32	275.5	32.62	226.6	14.74	213.7	16.05	-	-	196.8	16.14
atom x position	61.10	23.82	31.01	-4.8E-02	4.6E+02	107.7	7.07	84.3	21.18	76.0	7.88	78.8	7.61	67.3	12.88	68.6	9.07
atom y position	45.90	23.32	26.99	3.7E-02	2.1E+03	107.7	7.08	65.9	30.76	60.4	6.97	56.4	6.31	47.0	10.49	46.9	7.73
atom z position	61.68	23.84	27.48	9.1E-05	4.6E+02	107.7	7.46	94.6	19.86	82.6	9.00	86.1	8.25	75.7	13.80	78.2	9.93
atom y velocity	64.65	23.87	30.96	1.5E-01	1.4E-01	107.7	7.30	95.7	19.88	93.8	10.07	99.1	9.65	84.8	14.93	87.6	9.92
atom temp	64.91	23.94	27.41	3.0E-03	7.1E+03	107.7	6.69	95.7	19.76	91.6	10.27	95.9	8.34	84.6	15.02	84.6	10.31
atom energy	3.45	18.57	21.79	-3.6E+00	-2.7E+00	107.7	7.15	77.9	38.59	74.1	7.98	71.8	7.01	60.8	12.66	60.5	8.30
lucy	61.39	24.38	31.09	-6.1E-02	1.2E+03	160.5	-	137.8	-	99.5	-	90.0	-	73.6	-	77.8	-
david1 <sub>norm</sub>	25.23	17.08	31.11	-4.4E+03	1.8E+03	322.5	-	144.9	-	155.7	-	163.4	-	108.6	-	131.9	-
torso	84.72	18.48	31.08	-2.7E+02	5.8E+02	1.9	-	1.7	-	1.5	-	1.5	-	1.3	-	1.3	-
rhl	71.90	20.14	25.99	1.5E+00	3.6E+02	8.4	-	7.1	-	5.8	-	5.6	-	4.7	-	4.8	-

Table 1. Compression results for the Miranda (m2d, m3d, M3d) structured grids, the atom point set, the lucy and david triangle meshes, and the torso and rhl tetrahedral meshes. All data but M3d is represented in single precision. The [ILS2005] scheme operates on single precision only, hence the missing values. For the meshes we report only the compressed size of vertex coordinates; timings are dominated by connectivity coding, and are hence excluded. The range measures the (logarithm of) the number of floating-point values between min and max. Note that the first-order entropy is limited by the number of samples in a data set.

tation.) Arguably such data sets should use an integer rather than floating-point representation, although for simplicity or other reasons it is common practice to use floating-point. Contrary to [16], which entropy codes all bits of the residual, our new coder sacrifices such potential compression gains for speed by storing these repeated low-order bits in raw and uncompressed form. However, the massive data sets from scientific simulation that motivated our work on high-speed compression, as well as our tetrahedral meshes, rarely exhibit any significant low-order redundancy, as also evidenced by our results.

### 5.1.1 Lossy Compression

Fig. 3 shows that our scheme gracefully adapts to decreasing levels of precision when discarding the least significant mantissa (and eventually exponent) bits. For  $n$  bits of precision, the schemes [7, 22] require  $\log_2 n$  bits to code the number of leading zeros, whereas our scheme exploits the combination of low entropy in the leading-zero count and the elimination of the low-order bits that are most difficult to predict and compress.

### 5.2 Compression Speed

Fig. 4 shows the speed of compressing from memory to disk, including disk write time. (Because the throughput is only 20% less than our method, its decompression speed is similar to its compression speed.) We also include the raw I/O performance of dumping the data uncompressed using a single `fwrite` call. Timings

correspond to the median of five runs. Whereas our compressor is slightly slower than the less effective compressors [7, 22], it is nearly twice as fast as [16] while producing similar compression rates. However, in more I/O-intensive scenarios, such as in massively parallel simulations dumping data to the same file system (as is common), the improved compression of our method over [7, 22] results in a net gain in effective throughput. We integrated our compression code with Miranda's dump routines and ran performance tests on 276 nodes of LLNL's MPP supercomputer. Achieving on average a lossless reduction of 3.7 on 75 GB of data dumped, the overall dump time was reduced by a factor of 2.7 over writing the data uncompressed.

### 5.3 Entropy Coding

We compared the raw throughput of our range coder and Schindler's [23] by (1) passing raw bytes through it with no compression and (2) entropy coding byte sequences. In both cases, the source data was the uncompressed floating-point data used in our experiments. Timings show that our coder is 40% faster for raw transmission and 28% faster for entropy coding. Meanwhile, the inefficiency of our coder due to loss of precision and range reduction is only 26 bytes of overhead for 1.5 GB of coded data. Its raw throughput is only 20% less than an `fwrite` call, while its entropy coding throughput of 20 MB per second, which includes probability modeling and I/O time, compares favorably with state-of-the-art entropy coders [25].

Lindström et al. [2006]

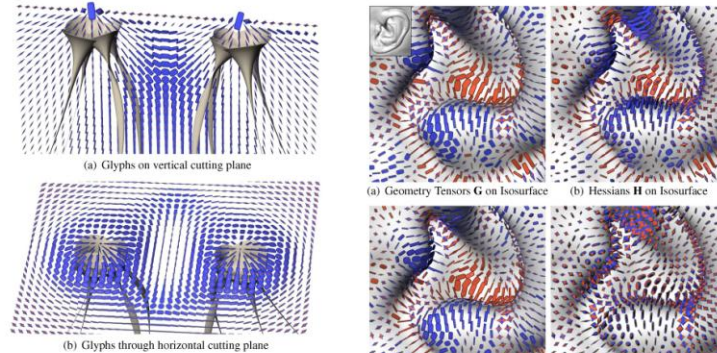


Fig. 8. Glyphs in the double point load stress tensor field reveal the minor eigenvector along which hyperstreamlines [7] are traced (a), and the variation in stress with distance from the load (b).

volume rendered [32], but its eigenvectors are commonly used in non-photo-realistic rendering, e.g. curvature-based strokes [11, 14, 19]. Inspecting geometry tensors could help debug an NPR method giving unexpected results in an unfamiliar dataset. Fig. 9(a) visualizes geometry tensors  $G$  on an isosurface (sampled by a particle system [39]) of an ear from the Visible Human male CT scan. Variations in surface curvature are reflected in the new glyphs: convex (blue circles), concave (orange circles), and saddles (orange and blue stars). For comparison, Fig. 9(b) shows the full Hessian  $H$  from which  $G$  was computed.

The new glyphs may also have a role in visualizing the tensor ingredients of image analysis methods such as edge detection. One edge definition is zero-crossing on the second directional derivative along the gradient direction,  $f'' = \mathbf{n}^T \mathbf{H} \mathbf{n}$ . This edge surface is sampled by a particle system [33] in Fig. 9(c), showing the Hessians at the edge locations, and revealing close similarities with the geometry tensors on the isosurface in Fig. 9(a), indicating that one of the Hessian eigenvalues is near zero even though this is not part of the edge definition. Another edge definition is the zero-crossing of the Laplacian  $\nabla^2 f = \text{tr}(\mathbf{H})$ , and Fig. 9(d) illustrates the difference between the Hessians on this surface and those in Fig. 9(c). The consistently gray glyph halos in Fig. 9(d) indicate that these are traceless tensors.

As a demonstration of the glyphs in a 2-D visualization, Fig. 10 visualizes a cross-section of a simulation of jet flow rightward into a steady medium, causing turbulence. Glyphs of rate-of-deformations tensors document how an infinitesimal volume is stretched or compressed as it moves along the flow. A backdrop of line integral convolution [4] (with contrast modulated by velocity) provides visual context. Fig. 10(a) uses the exponentially-scaled ellipses of [34] to map tensors with negative eigenvalues to positive-definite tensors suitable for ellipsoid visualization. When the absolute difference between eigenvalues becomes too large, these glyphs can become so stretched that they overlap each other and extend over a significant portion of the domain, undermining the locality normally enjoyed by glyphs. Such stretching also reduces the visual presence of the needle-like glyphs for tensors with larger norms, contrary to scale preservation (6). Fig. 10(b) uses our superquadratic glyphs with  $s(\|\mathbf{D}\|) \approx \|\mathbf{D}\|$ . The aspect ratio reflects the relative eigenvalue magnitudes, the size correctly indicates the tensor norm, and pointed glyph shapes clearly communicate eigenvector directions. With compression of scale variation

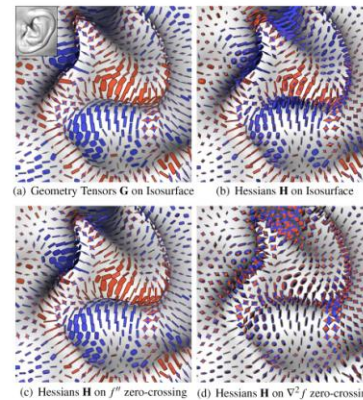


Fig. 9. Visualization of geometry tensors (a) and Hessians associated with isosurfaces (b) and two different definitions of edges, zero-crossings of the second-directional derivative (c) and the Laplacian (d). These results use  $s(\|\mathbf{D}\|) \approx \|\mathbf{D}\|^{1/2}$  in (6).

$s(\|\mathbf{D}\|) \approx \|\mathbf{D}\|^{1/2}$ , Fig. 10(c) better shows the directional patterns where the tensor norm is low. Colormapping the rate-of-deformation tensor trace with glyph halos highlights the regions of over-all stretching or compression, especially along the bottom edge of the domain.

Finally, Fig. 11 demonstrates how our new glyph performs traceless tensor visualization, in a side-by-side comparison to the dedicated traceless NLCT tensor glyphs by Jankun-Kelly et al. [25]. Traceless tensors form a plane in eigenvalue space, and we are visualizing samples from a square within this plane, centered around the zero tensor (cf. Fig. 4(c)). Unlike the traceless glyph, which maps tensor norm to glyph sharpness, our glyph expresses norm by its overall scale  $s(\|\mathbf{D}\|) \approx \|\mathbf{D}\|$ . Consequently, the traceless glyph requires pre-specification of maximum eigenvalue magnitudes (which are mapped to perfect sharpness), while our glyph can be used without such prior information. Another notable difference is that limiting their glyph to traceless tensors allows Jankun-Kelly et al. to make use of parts of the superquadratic shape space – including cylinders and boxes – that our approach sets aside for positive- or negative-definite tensors.

### 6 CONCLUSION

Visualization research has made significant progress in visualizing second-order tensor fields, but has mostly concentrated on the positive-definite case. Faced with indefinite tensors, a frequent strategy is to map them to positive-definite tensors prior to visualization [34, 22, 21, 52, 33]. Even when bijective mappings are used (so mathematically, no information is lost), such mappings still visually obscure the difference between positive and negative eigenvalues, which is a fundamental qualitative aspect in various applications.

Therefore, we propose an extension of a previous positive-definite tensor glyph [28] to the full space of symmetric second-order tensors. Our glyph emphasizes differences in eigenvalue sign in a way that, unlike the Reynolds glyph [18], prevents small eigenvalues from being occluded by larger ones. We also propose to use halos to ensure tensor glyph visibility even when one or more eigenvalues are near zero. Finally, we present a time- and memory-efficient implementa-

Schultz & Kindlmann [2010]

two cases was not a factor that could affect the outcome of this study negatively. One experimenter engaged in the demonstration of the table and in assisting the participant when needed. A second experimenter took notes and also documented the session by voice recording. A prepared interview guide was used. It included a set of predefined questions that covered various aspects of the design goals (section 5.1) and also a number of potential questions used to prompt the participant to “think aloud” when needed. Both experimenters engaged in the conversation and made sure that all questions in the interview guide were covered by the end of the session. Some questions were discussed while participants worked on the cases and some were reviewed afterwards.

After using the table participants completed a subjective satisfaction questionnaire. The responses were given on a 5-point rating scale: *Strongly unfavorable* (1), *Unfavorable* (2), *Unsure* (3), *Favorable* (4), and *Strongly favorable* (5). The questionnaire covered the following

1. **Overall impression:** The overall impression of the table.
2. **Willingness to use:** Whether the orthopedic surgeon would like to use the table in their daily work.
3. **Efficiency for planning:** Whether using the table would save time for pre-operative planning.
4. **Efficiency for surgery:** Whether using the table pre-operatively would save time during actual surgery.
5. **Quality of work:** Whether using the table would improve the quality of clinical work.
6. **Learnability:** Ease of learning table usage for a novice user.
7. **Interaction:** Ease of interacting with the table.
8. **Benefits over static imaging:** Whether interactive 3D imaging is superior to the series of static 3D snap-shots used today.
9. **Collaboration:** Whether access to the table at work would facilitate collaboration between several people.
10. **Similarity to real situation:** Whether the similarity to a real situation (patient lying on a table) facilitates insights and decision making.
11. **Recommend to others:** Whether the orthopedic surgeon would recommend colleagues to use the table at work.

### 7 RESULTS

The user study proved effective for the objective of collecting distinct and broad feedback from the orthopedic surgeons about how the visualization table would fit in their application domain. They did not consider the “think aloud” approach to be distracting from the evaluation tasks. The overall assessment from the surgeons is that the table would be useful in their clinical work. This is illustrated by the numerical ratings in the post-session questionnaire, see figure 12. Responses for the eleven statements has a group mean value<sup>1</sup> of 3.8 (two statements), 4 (two statements) and above 4 (seven statements) respectively, all corresponding to a clearly favorable rating. Statistical significance was however, not achieved but this is to be expected for this small study. There is only one example of a negative rating, one surgeon expressed moderate disagreement with the table’s potential to improve efficiency during surgery (specialist, age 50). There were three statements concerning general impressions of usefulness, *Overall impression*, *Willingness to use*, and *Recommend to others*, and in all three cases the study shows a strongly favorable mean rating of 4.4. Both the younger and less experienced participants and the older specialists contributed

<sup>1</sup>It can be discussed whether averaging in an ordinal scale is appropriate or conclusion is that in this case it is the best way to convey the results, in combination with the min-max measures in figure 12.

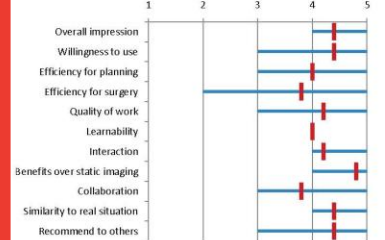


Fig. 12. The quantitative results of the user study questionnaire. Subjective satisfaction regarding use of the table was measured for 11 questions, see section 6. The 5-point rating scale ranges from *Strongly unfavorable* (1) through *Unsure* (3) to *Strongly favorable* (5). Vertical red bars denote the mean value and horizontal blue lines denote the full span of given ratings.

### 7.1 Ease of use and learnability

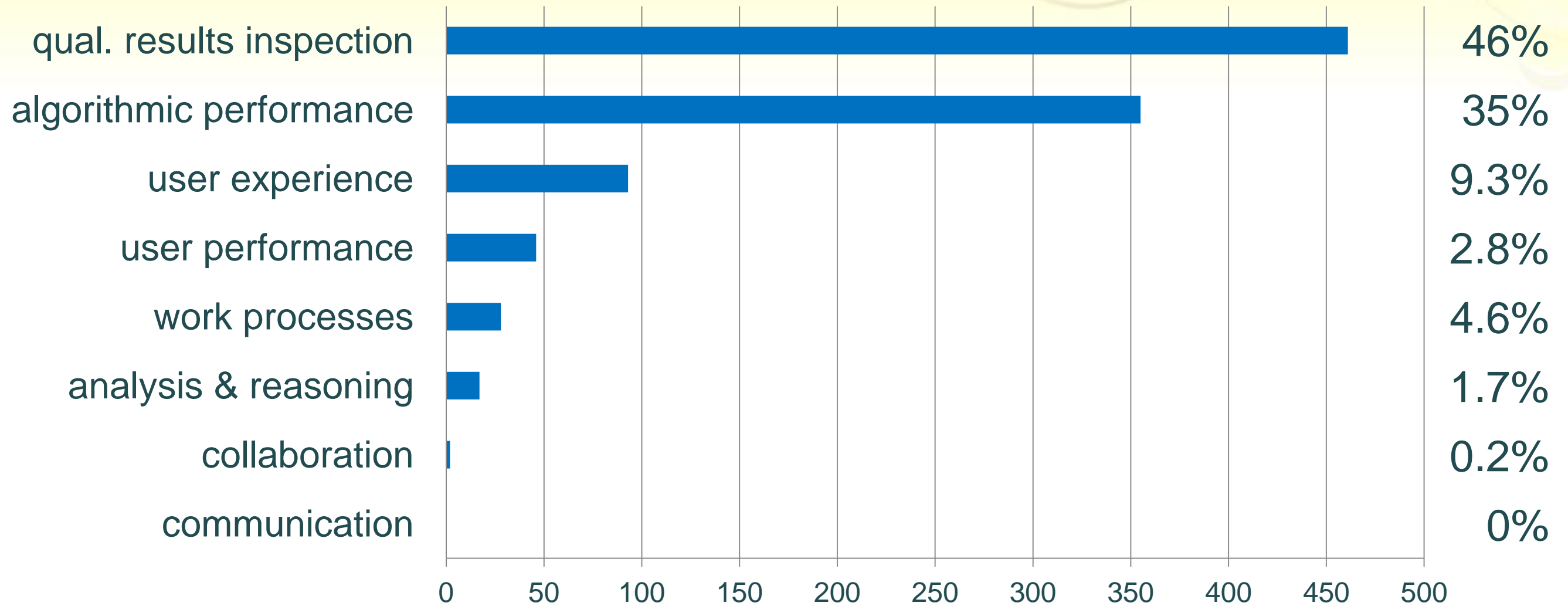
Low learning threshold and high usability were central objectives in the design of the system, reflected by design requirements R1-R5. Regarding the overall impression of the table all participants expressed positive statements. The interface was considered intuitive and convenient, and it was easy to learn how to use the basic functionality. The comments about learning threshold expressed an anticipation that novice users would quickly learn the basic functionality, although some of the more advanced functionality (activated via the pucks) would require some practice. All appreciated the clean interface with only a few visible GUI elements and emphasized the benefit and importance of the screen being focused towards visualizing the 3D image.

Regarding the interaction, the touch gestures were described as intuitive and straightforward to use, also for one of the participants who pointed out that he had never used a touch-controlled interface before (specialist, age 54). The surgeons were asked if they perceived the interaction as robust and responsive. They all concurred, through statements that the result of actions on the screen was what they expected and that they felt in control. Nobody mentioned that the level of precision provided by the touch technology and the RST interaction was insufficient or problematic. Even though the participants did not bring it up, the experimenters noted a few occasions of unintentional gestures due to holding the knuckles of inactive fingers too close to the surface. The typical effect was that panning occurred instead of an intended x-y-rotation, which the users dealt with by lifting the hand and reapplying the rotation gesture. For the additional MPR slice views it was commented that touch gestures were more efficient for transversal browsing than using a mouse.

The pucks were, in general, described as a convenient approach for reaching additional features. The interaction that caused some confusion was the advanced parts of the clip plane functionality, namely to control and understand slab clipping. The surgeons adopted the natural size zoom as an integral part of the toolset and no usability obstacles

Lundström et al. [2011]

# Results: evaluation scenarios

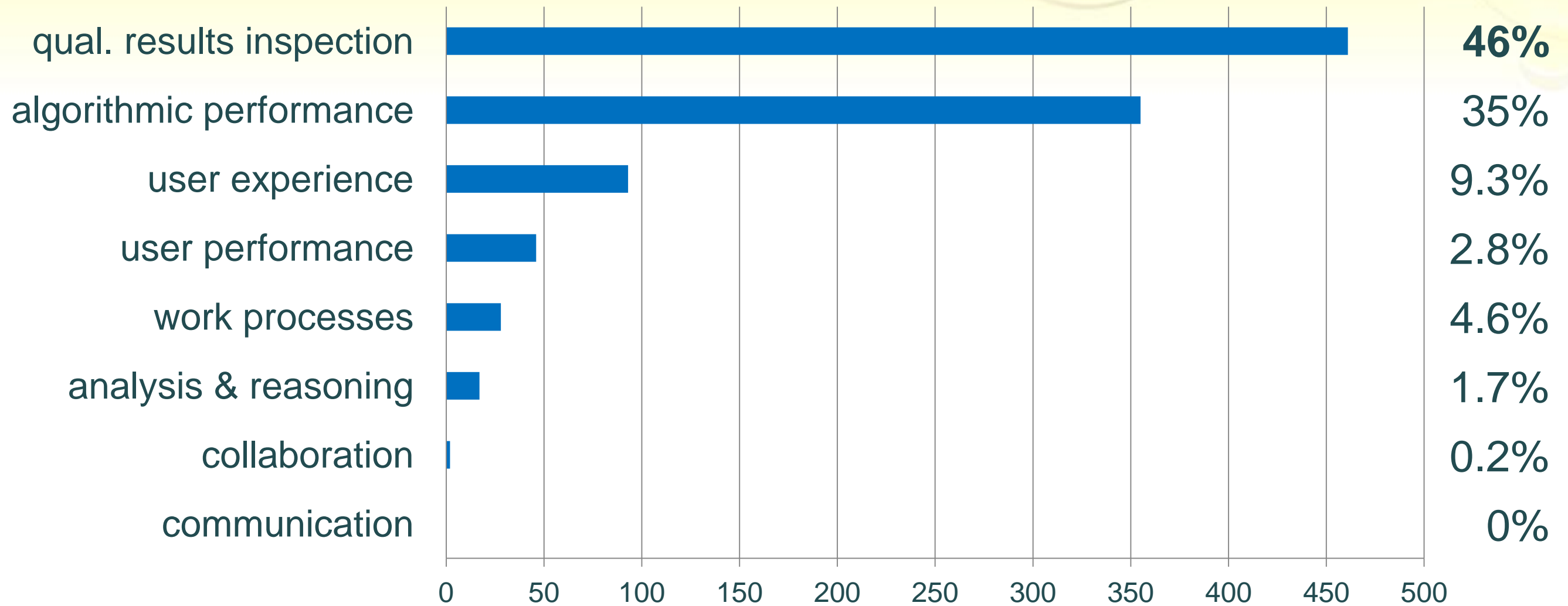


**IEEE Vis/SciVis: total numbers & percent of evaluation scenarios**





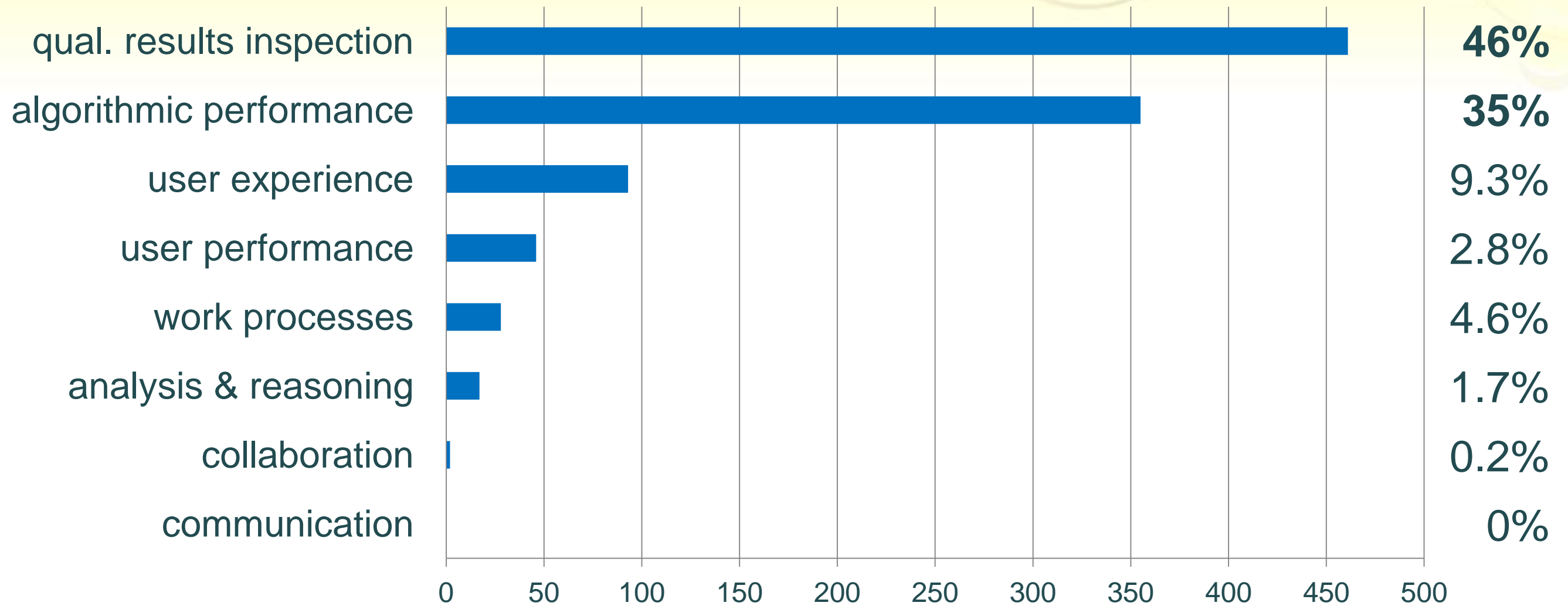
# Results: evaluation scenarios



**IEEE Vis/SciVis: total numbers & percent of evaluation scenarios**



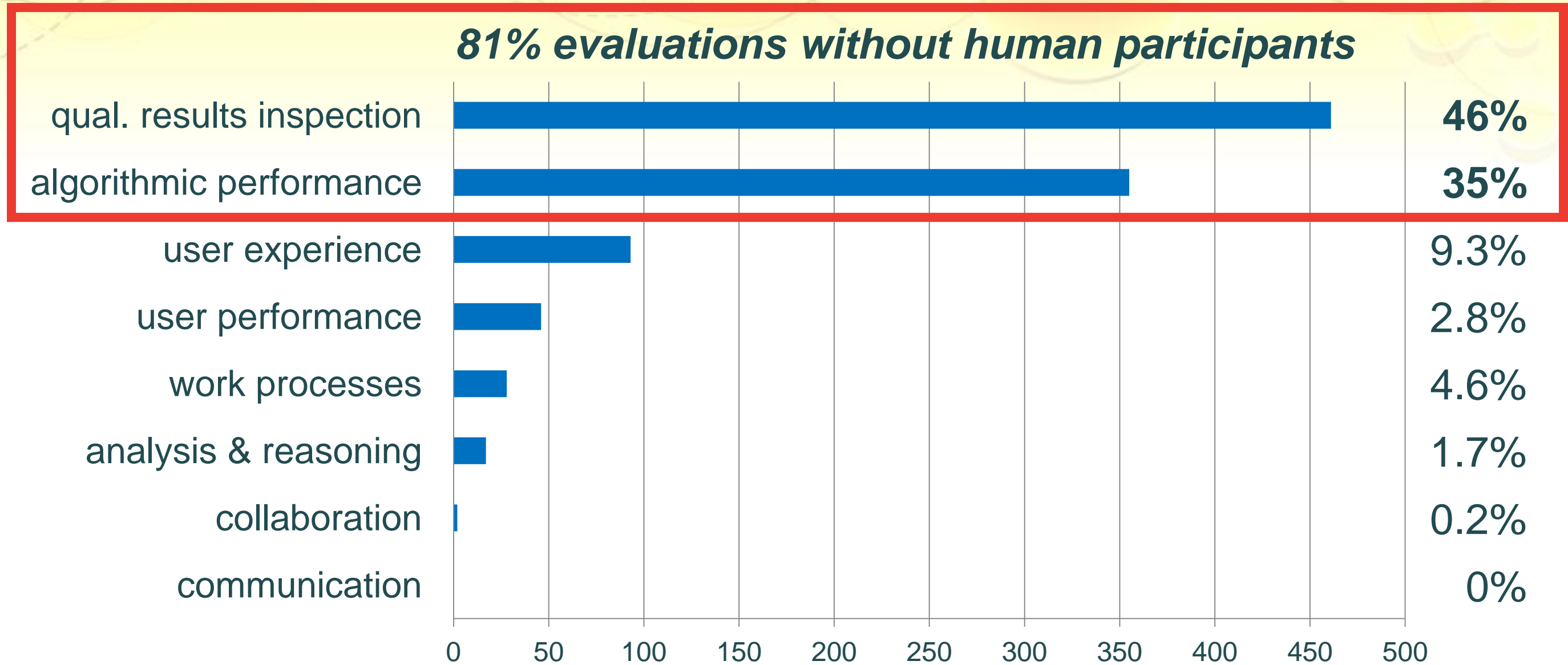
# Results: evaluation scenarios



**IEEE Vis/SciVis: total numbers & percent of evaluation scenarios**



# Results: evaluation scenarios

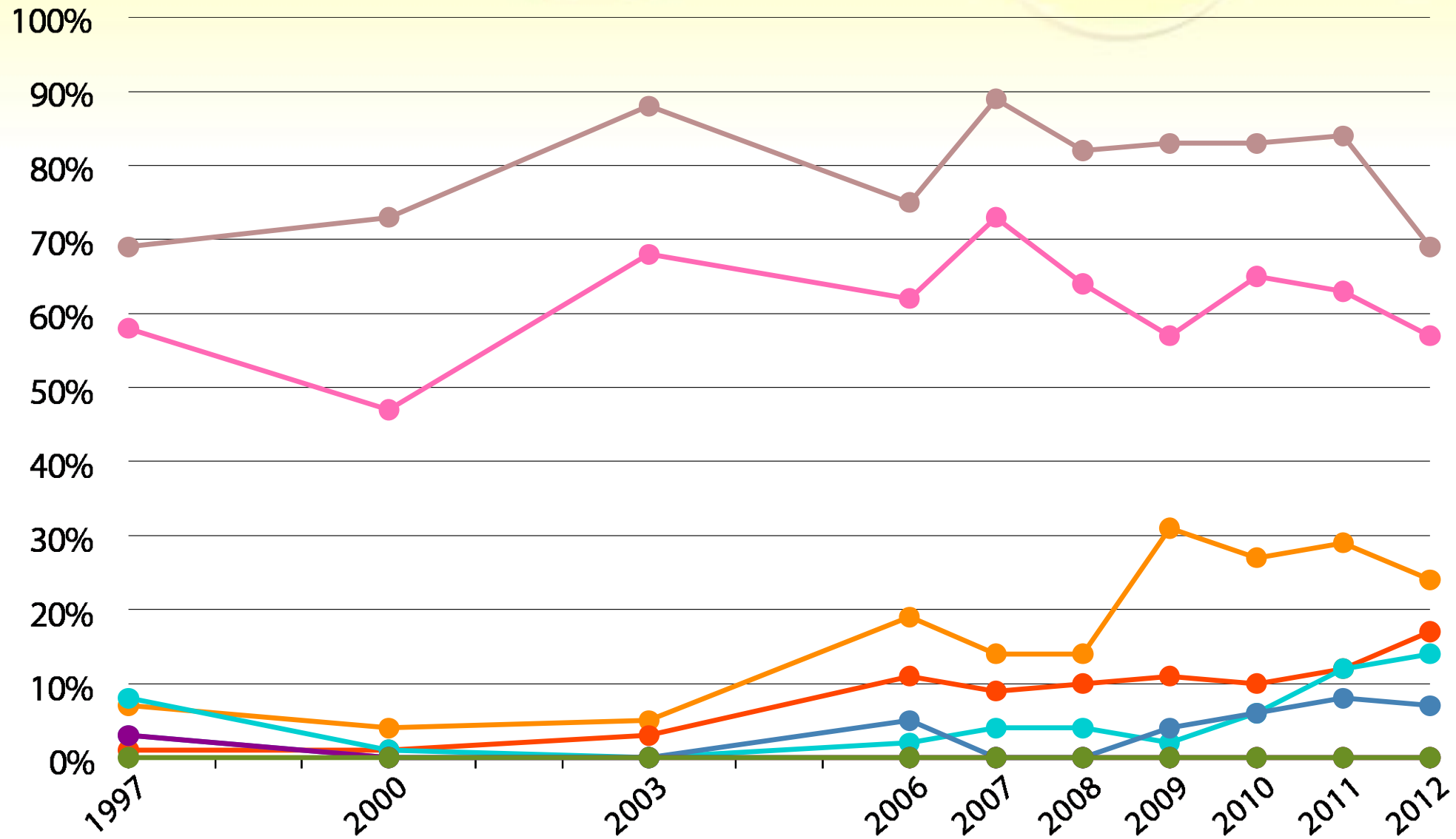


**IEEE Vis/SciVis: total numbers & percent of evaluation scenarios**

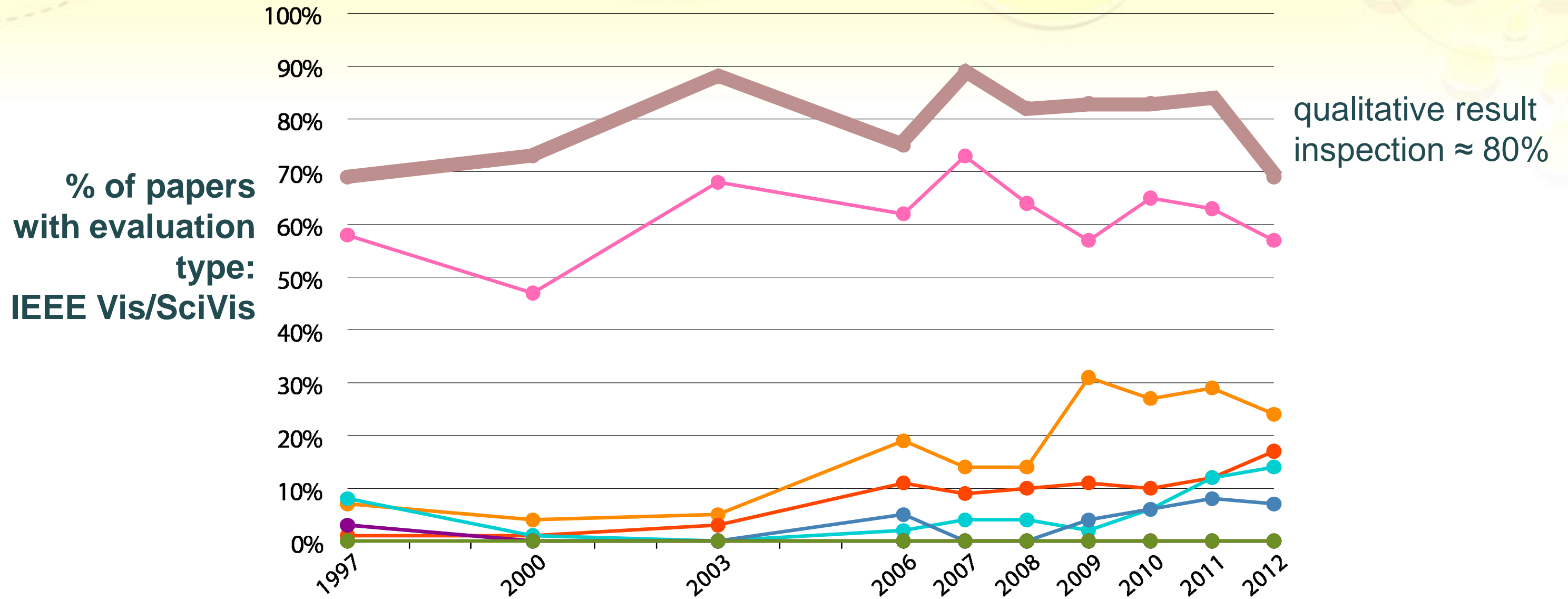


# Results: historical development

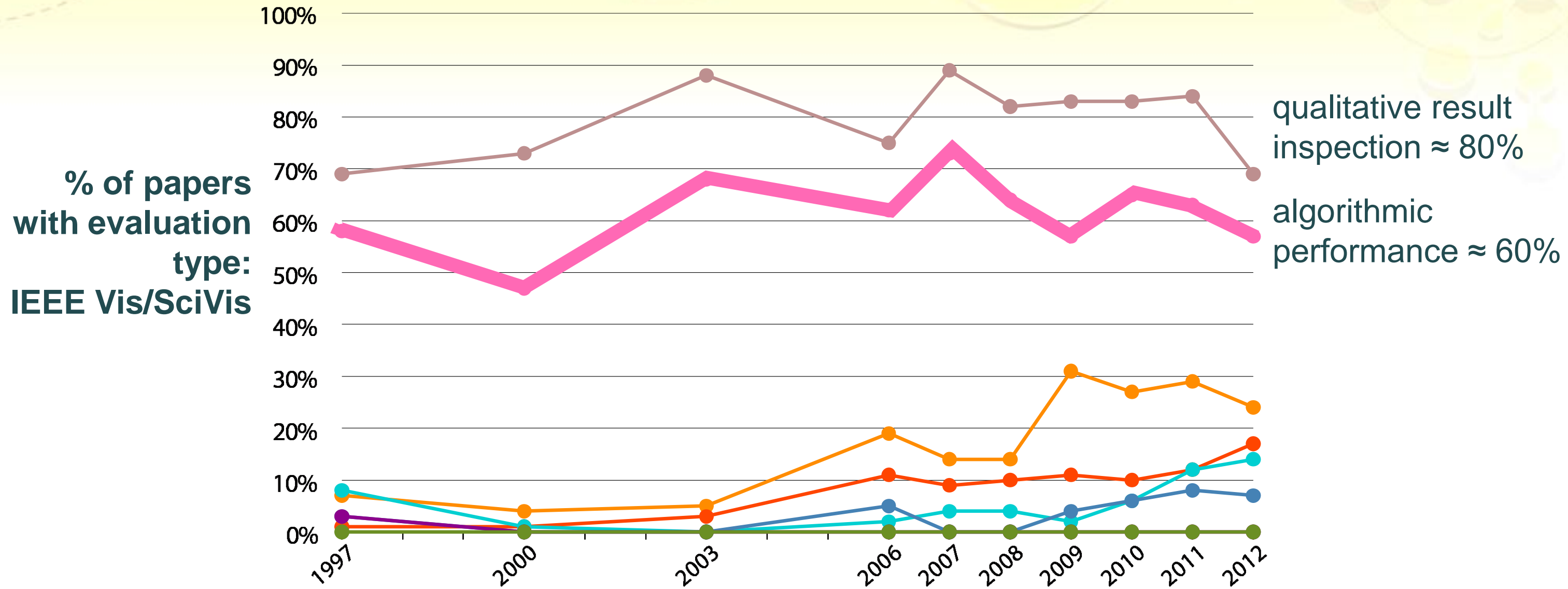
% of papers  
with evaluation  
type:  
IEEE Vis/SciVis



# Results: historical development

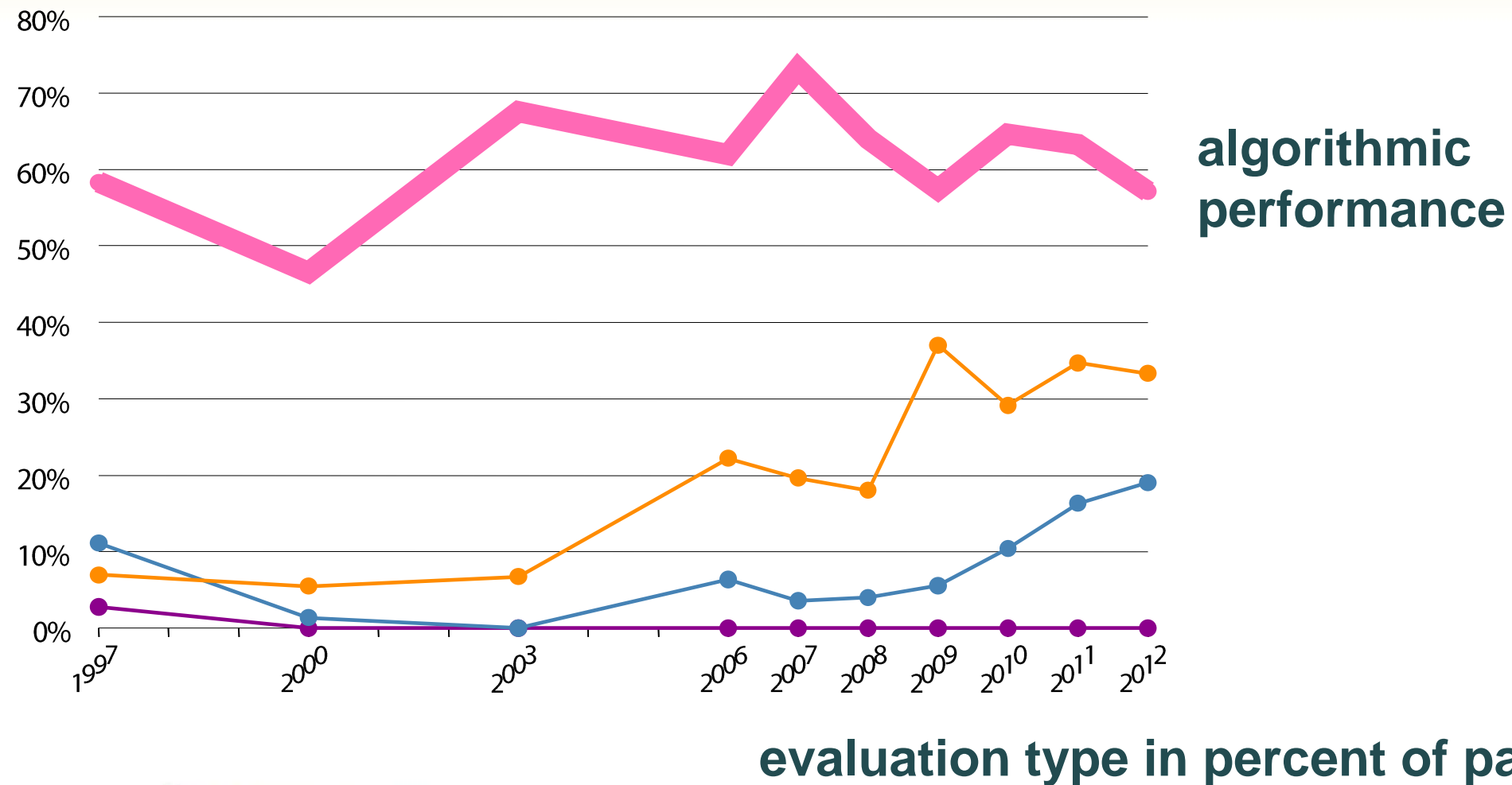


# Results: historical development



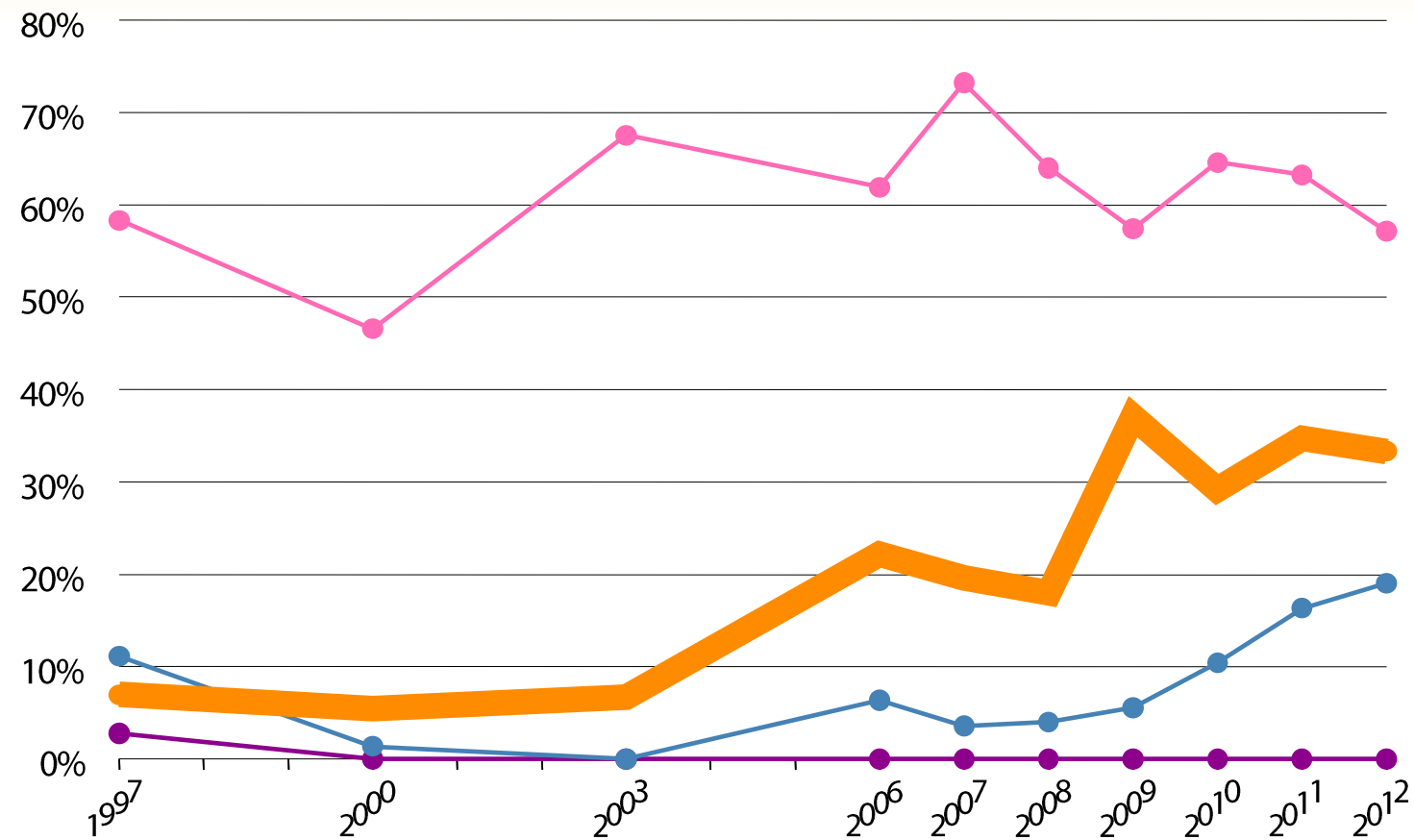
# Results: historical development

## IEEE Vis/SciVis



# Results: historical development

## IEEE Vis/SciVis



**user performance +  
user experience**

**evaluation type in percent of papers**





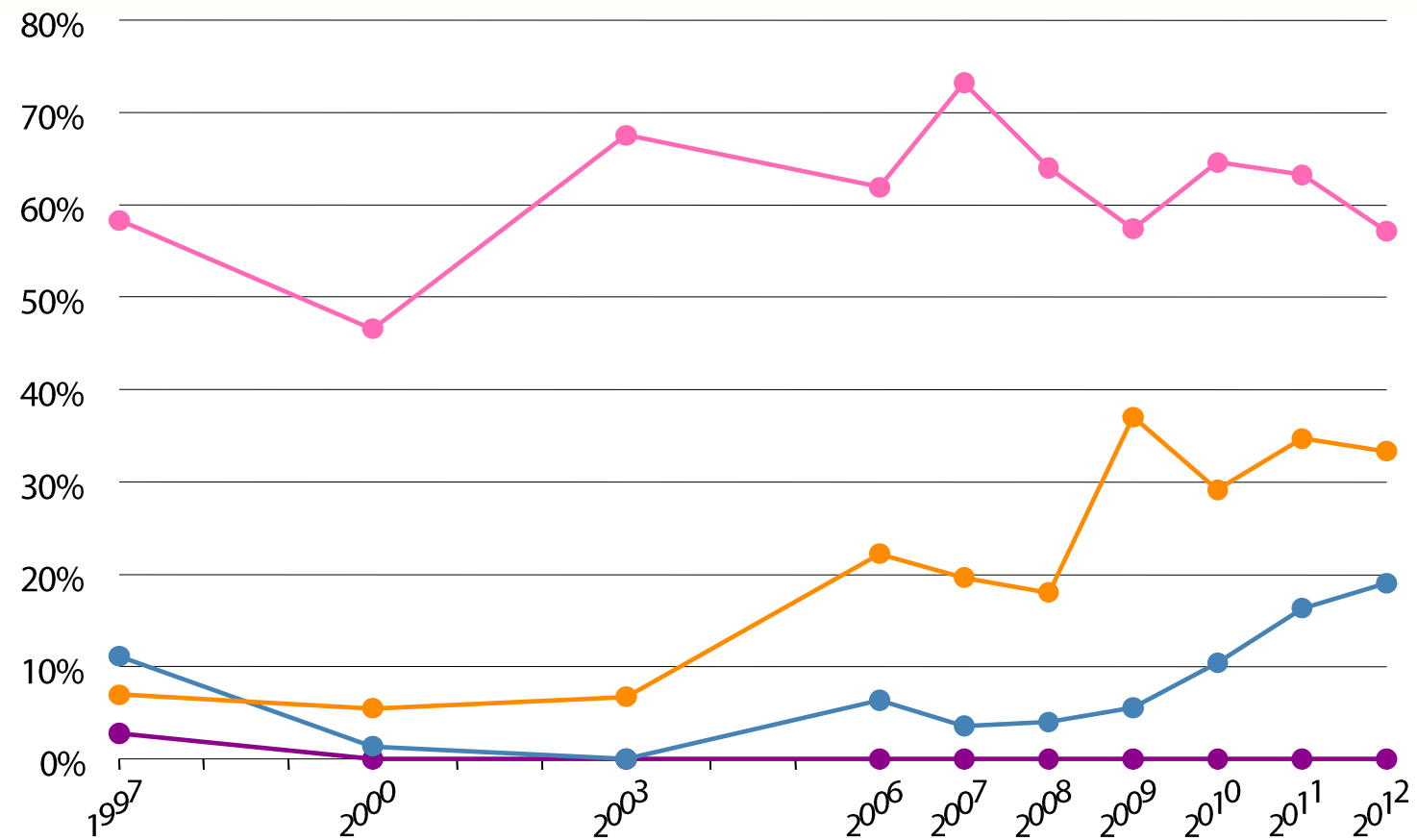
# Results: historical development

## IEEE Vis/SciVis



# Results: historical development

## IEEE Vis/SciVis

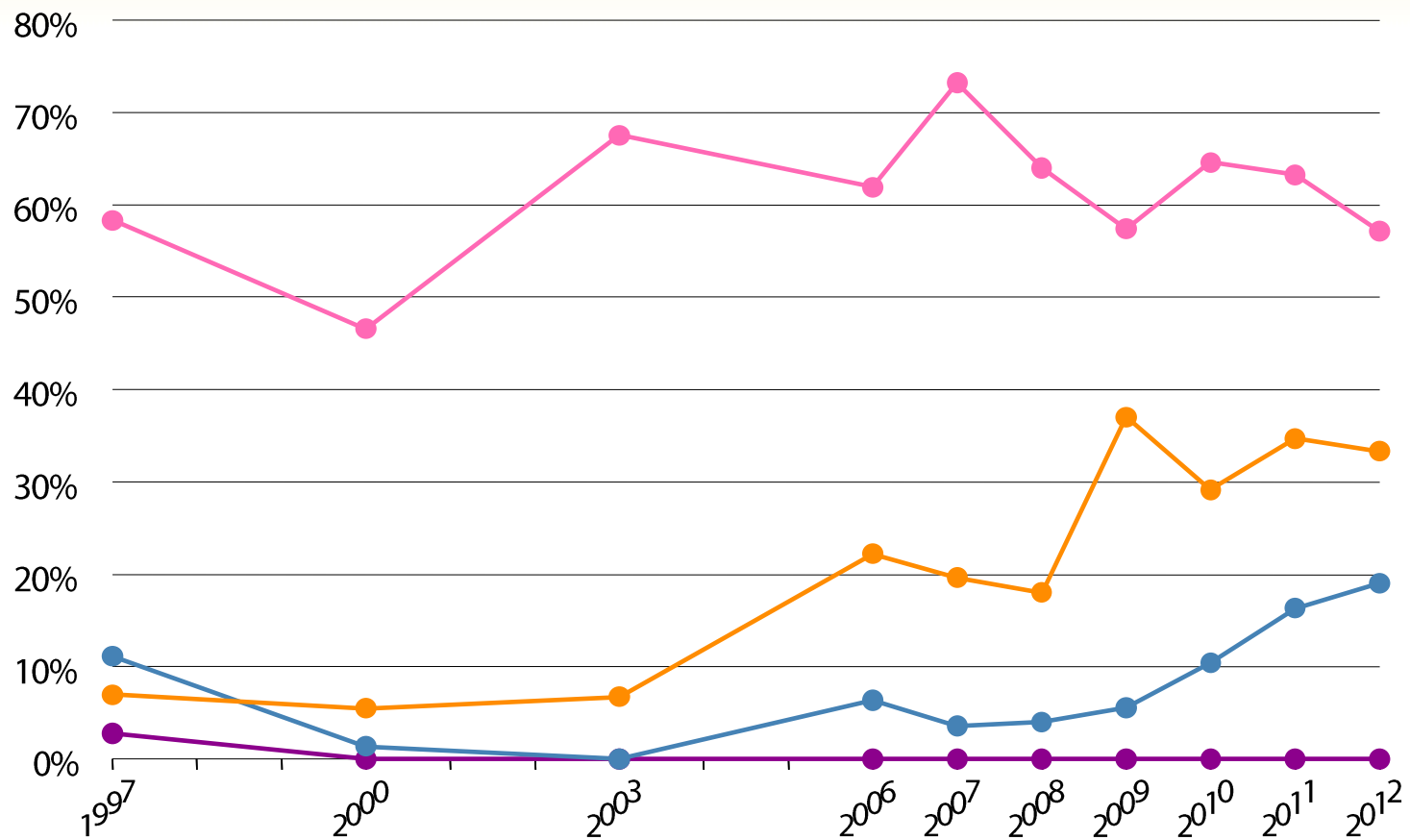


evaluation type in percent of papers

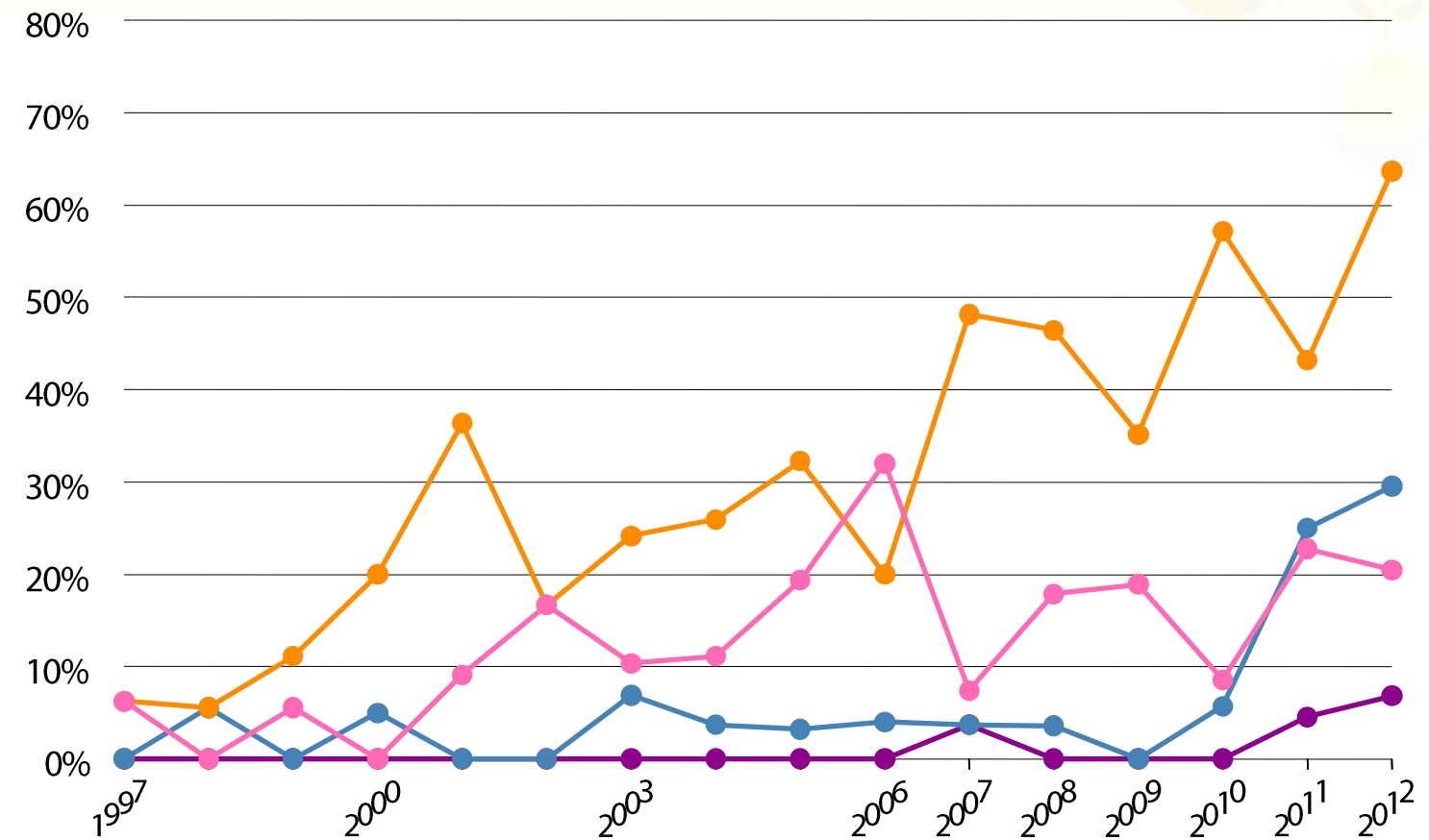


# Results: historical development

## IEEE Vis/SciVis



## IEEE InfoVis

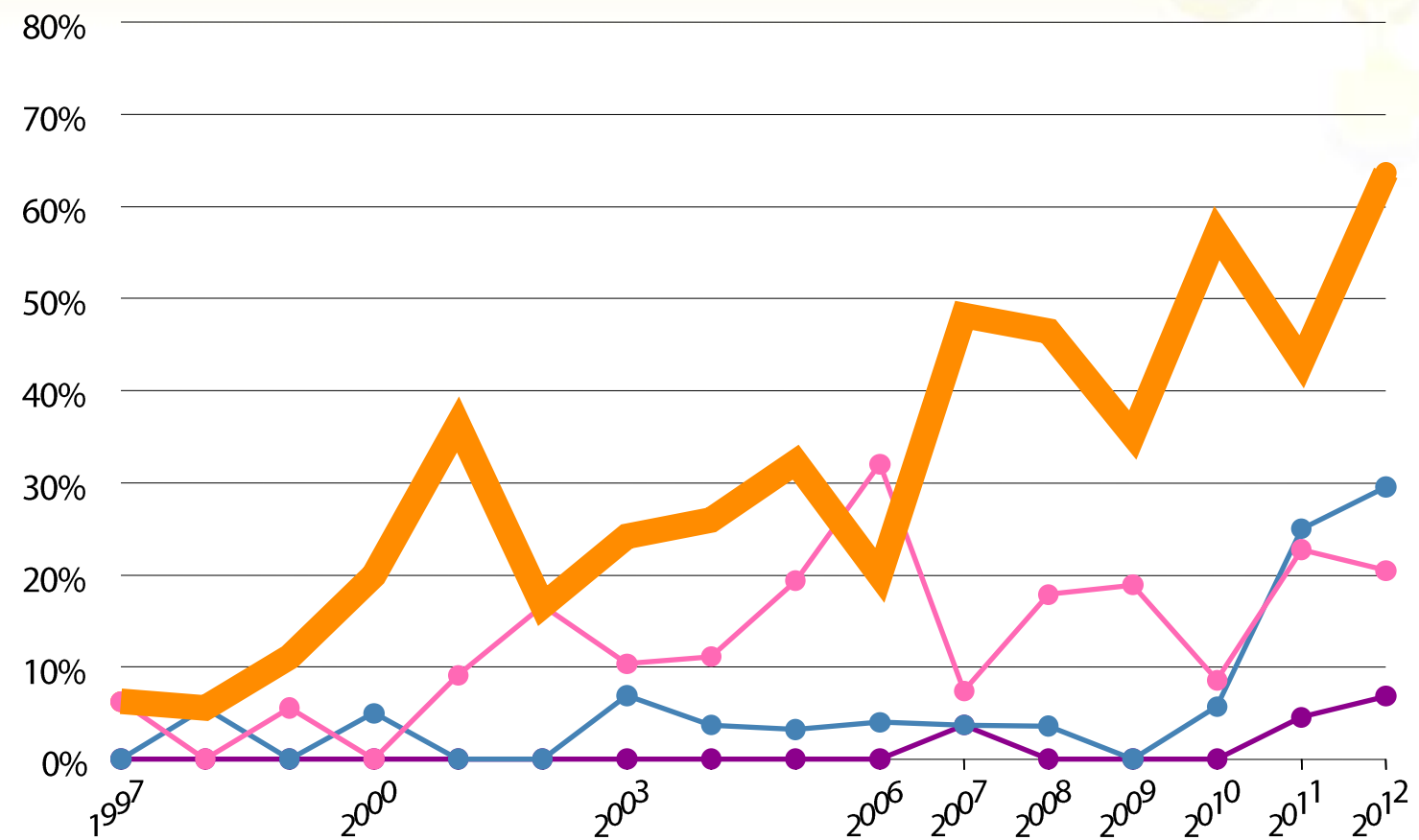
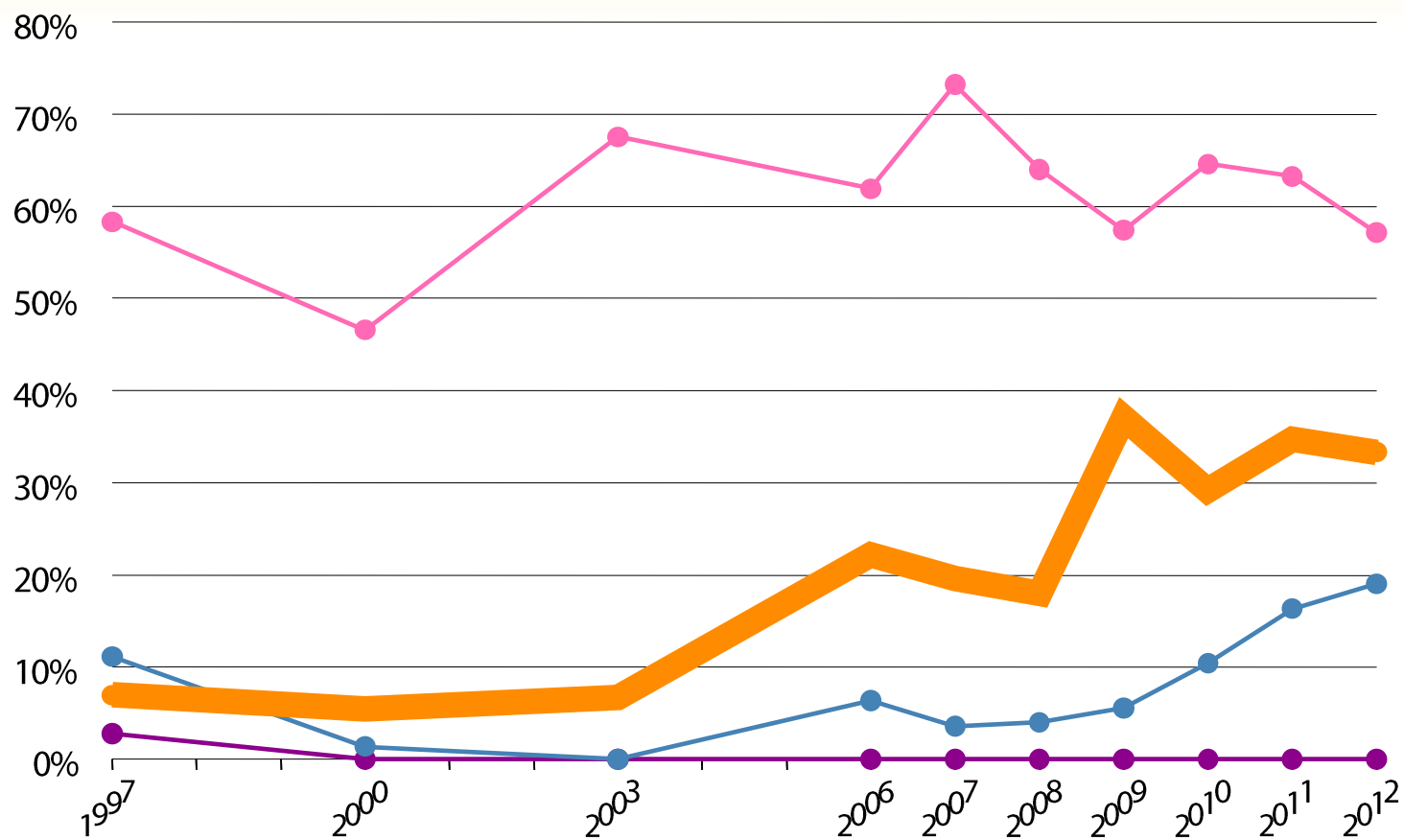


evaluation type in percent of papers



# Results: historical development

IEEE Vis/SciVis      user performance + user experience      IEEE InfoVis

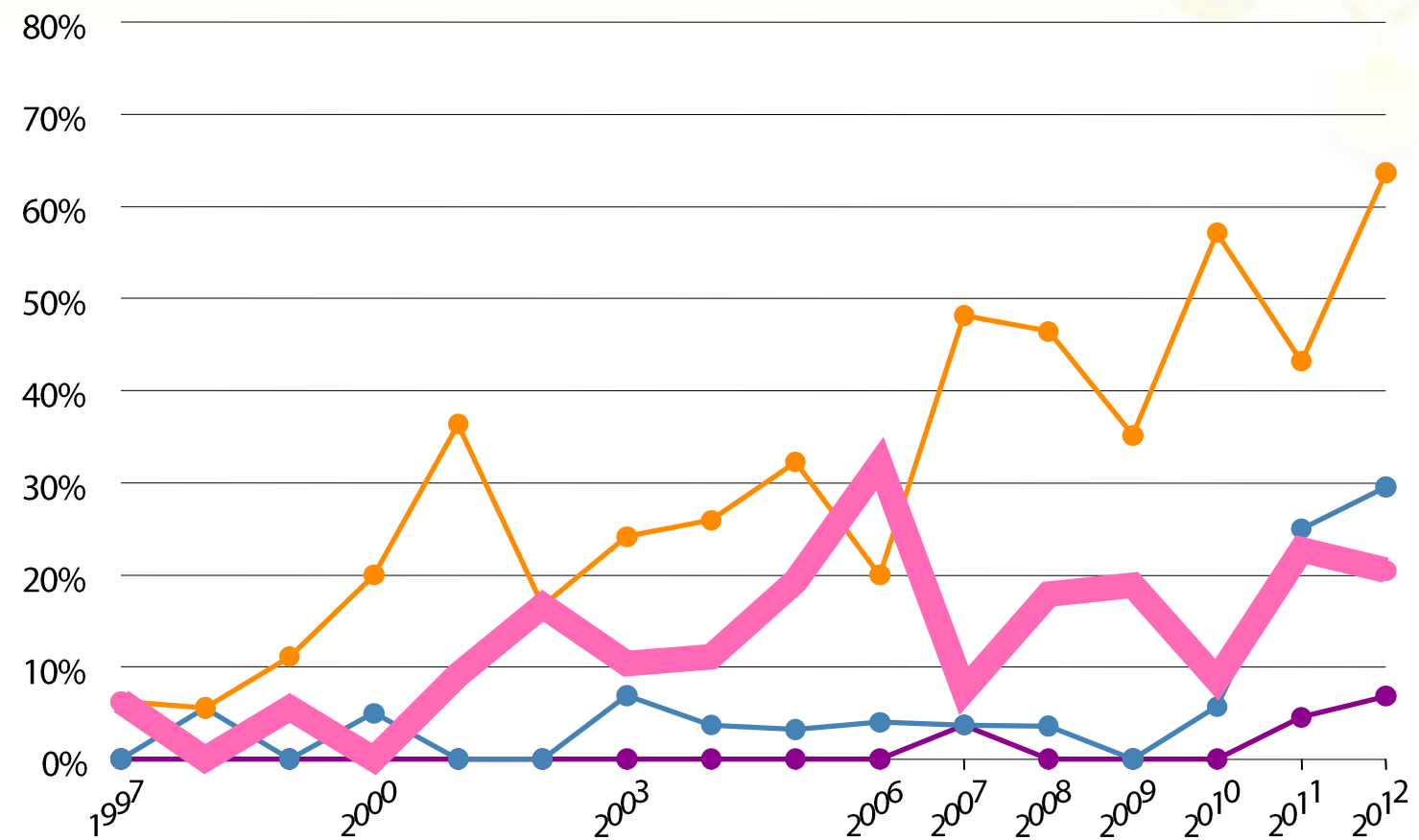
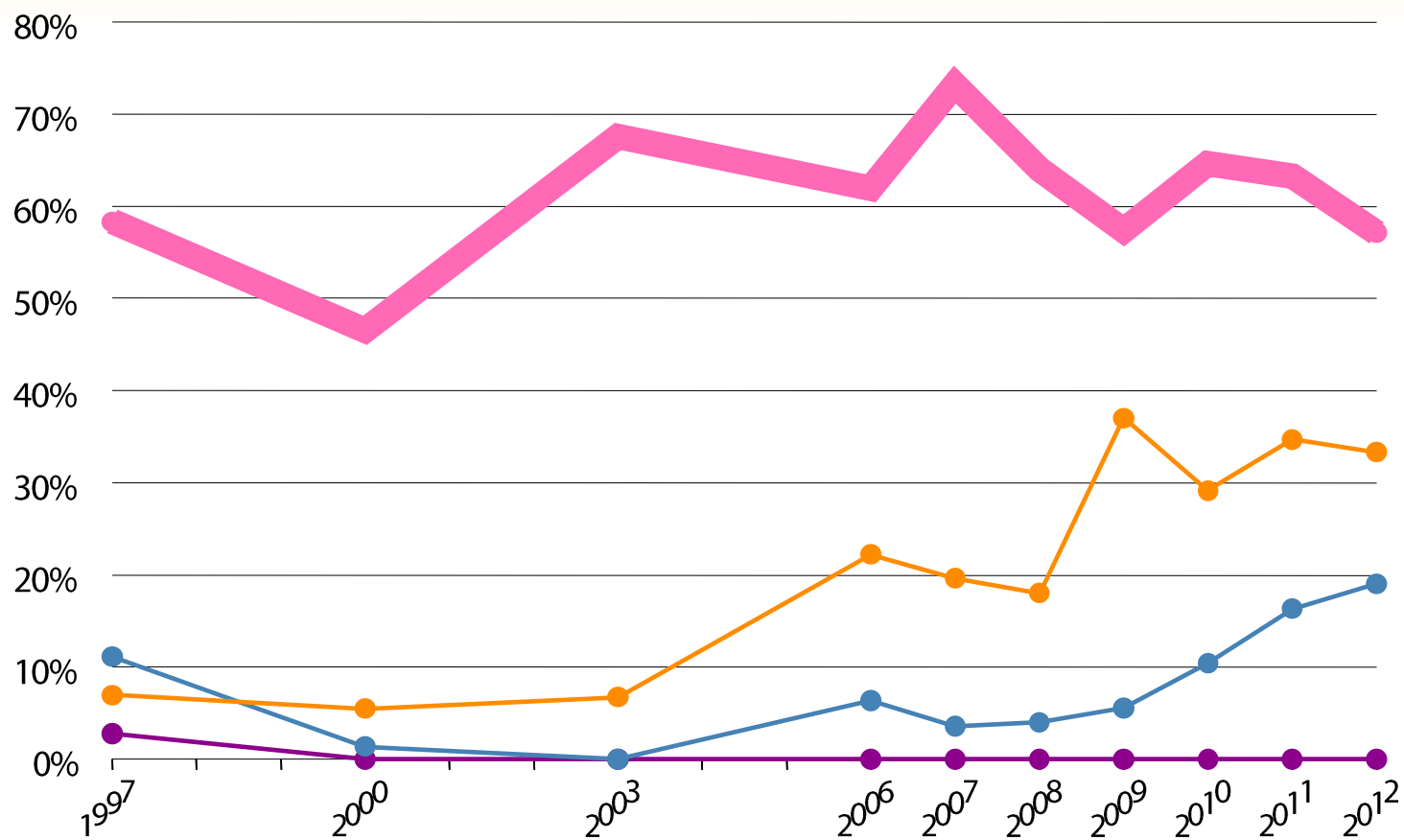


evaluation type in percent of papers



# Results: historical development

IEEE Vis/SciVis      algorithmic performance      IEEE InfoVis

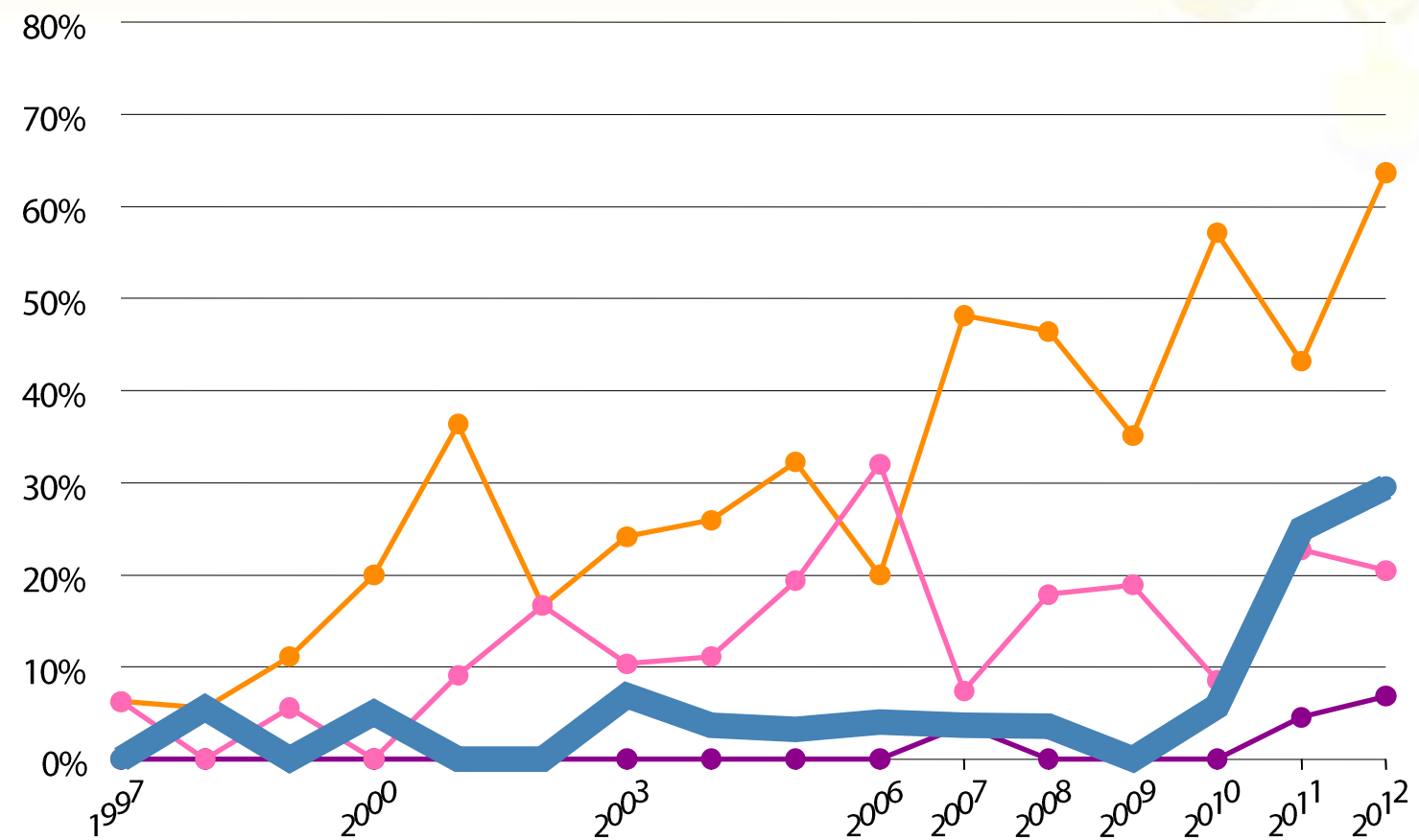
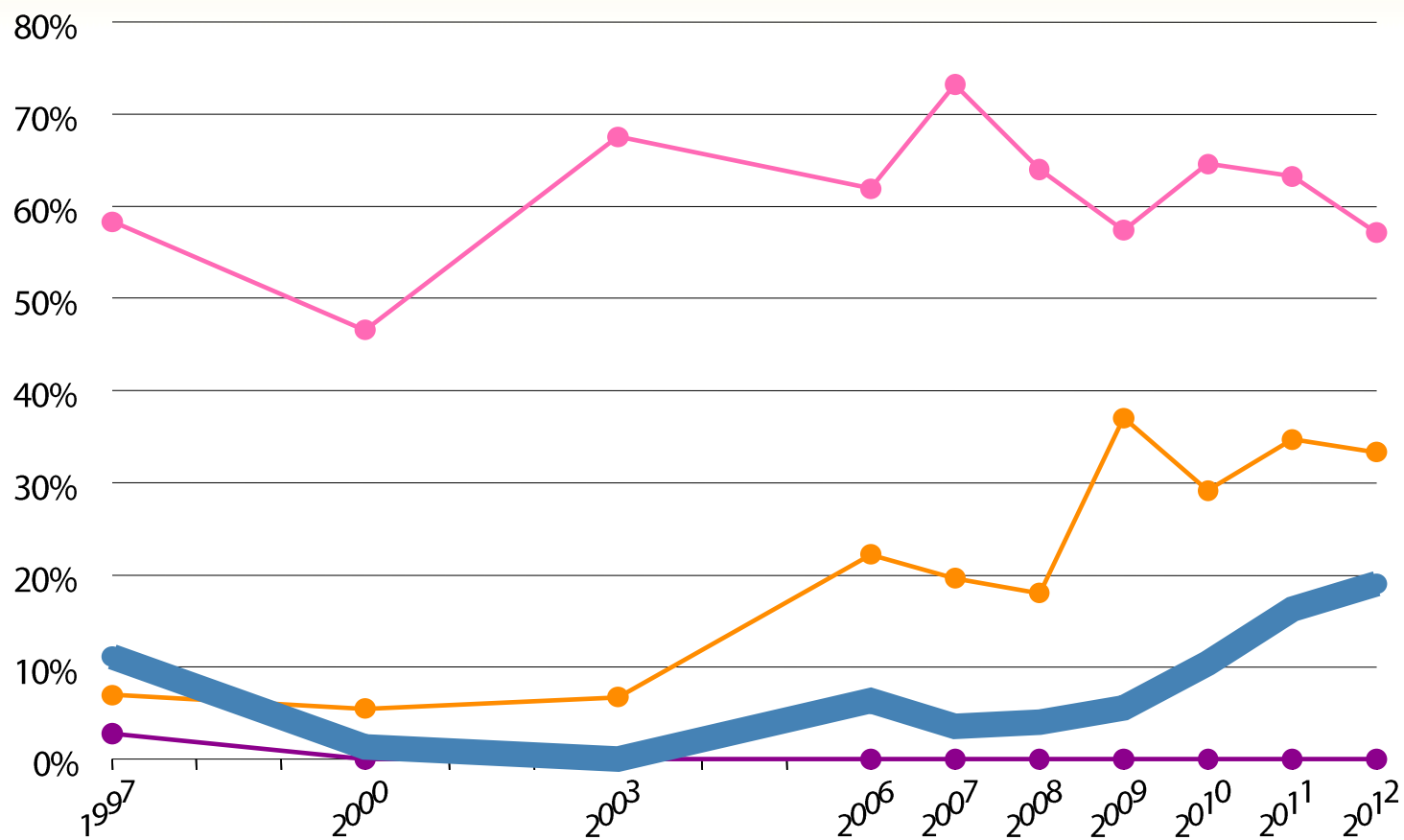


evaluation type in percent of papers



# Results: historical development

IEEE Vis/SciVis      work processes + analysis & reasoning      IEEE InfoVis

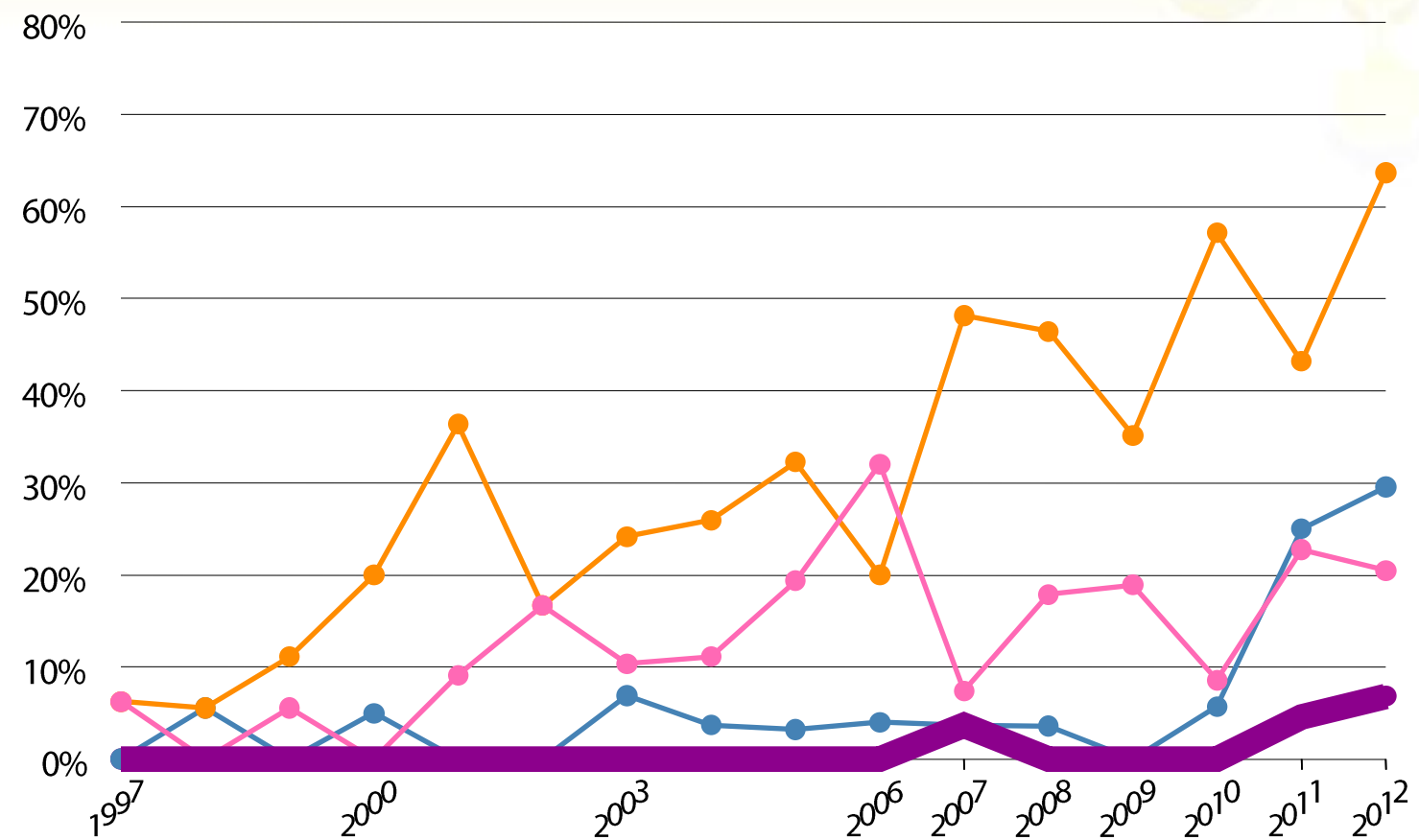
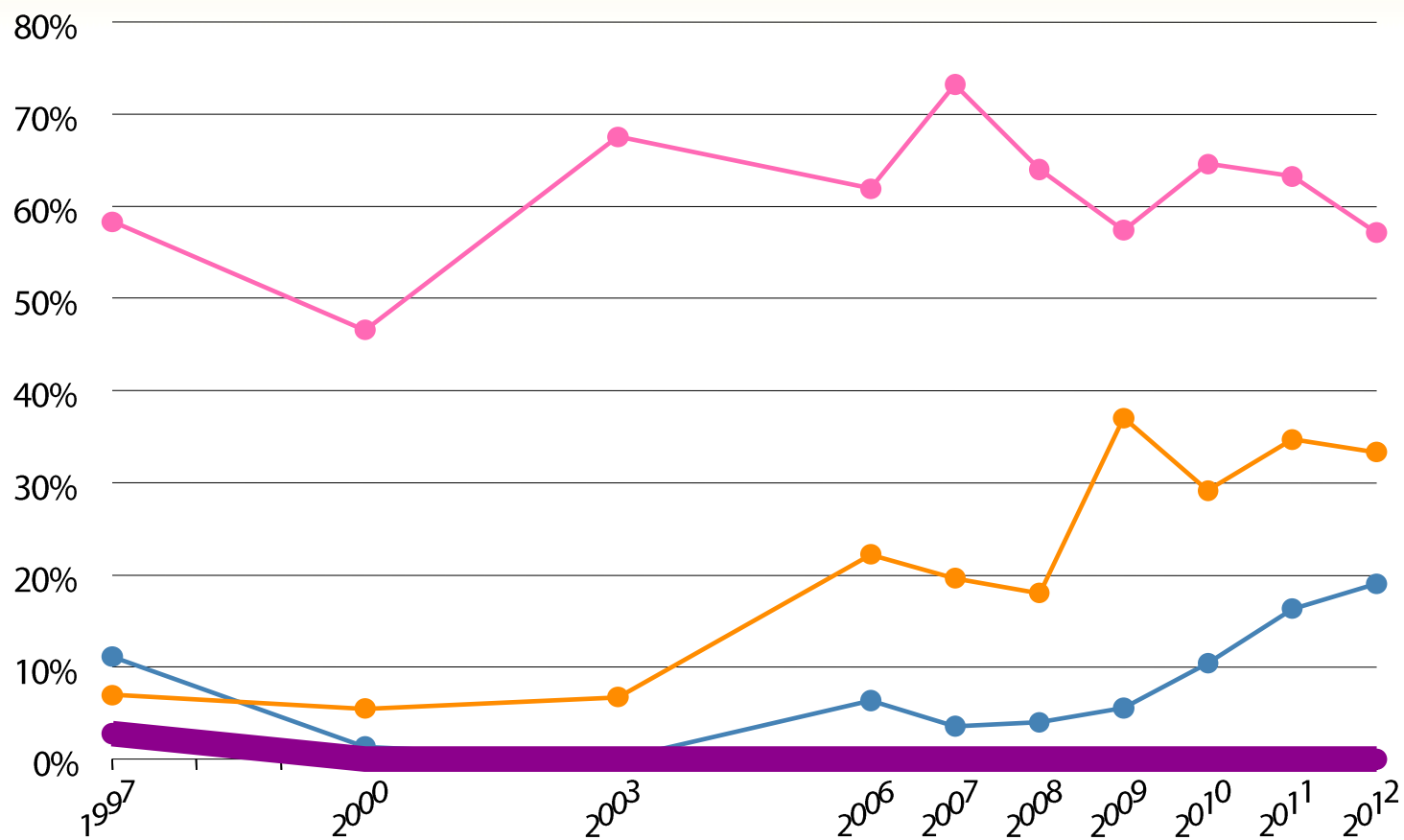


evaluation type in percent of papers



# Results: historical development

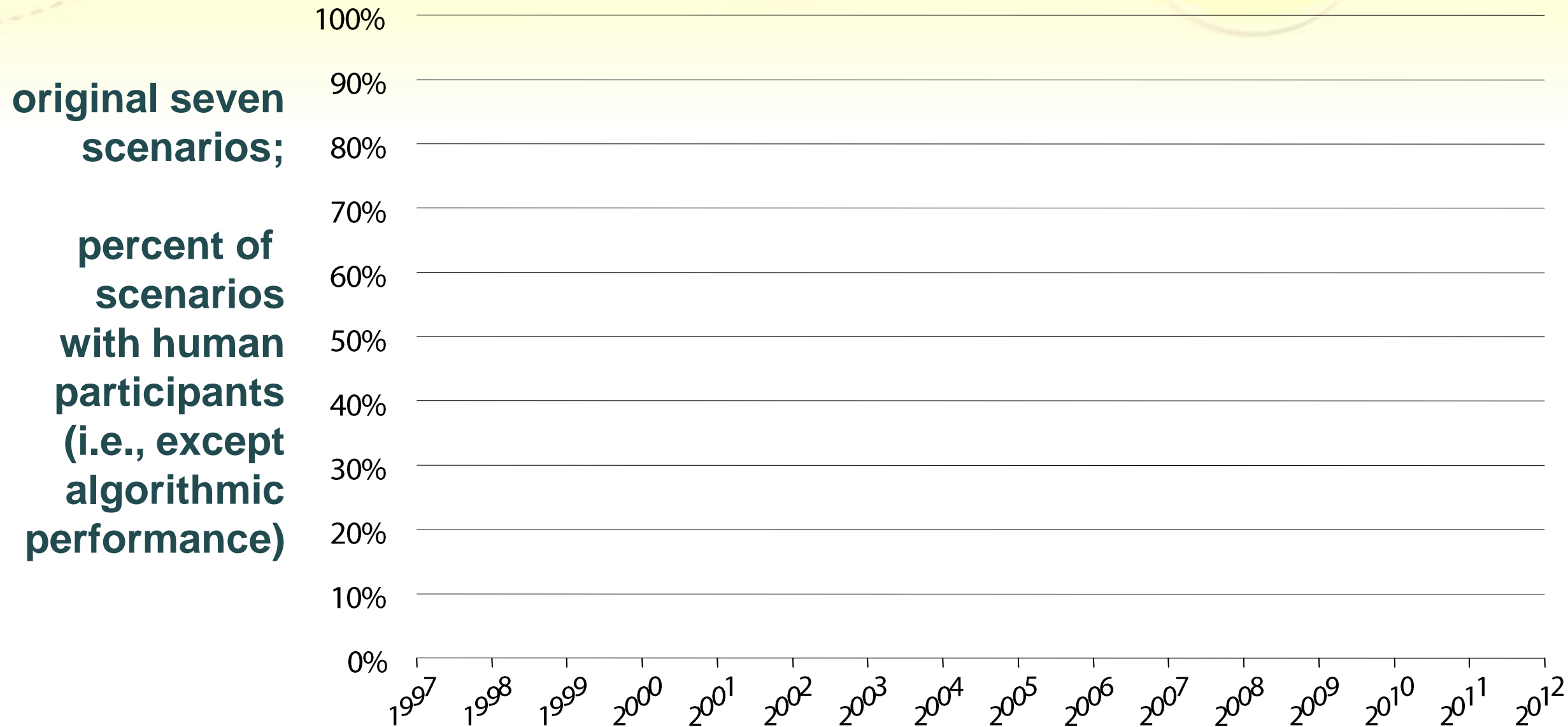
IEEE Vis/SciVis      collaboration + communication      IEEE InfoVis



evaluation type in percent of papers

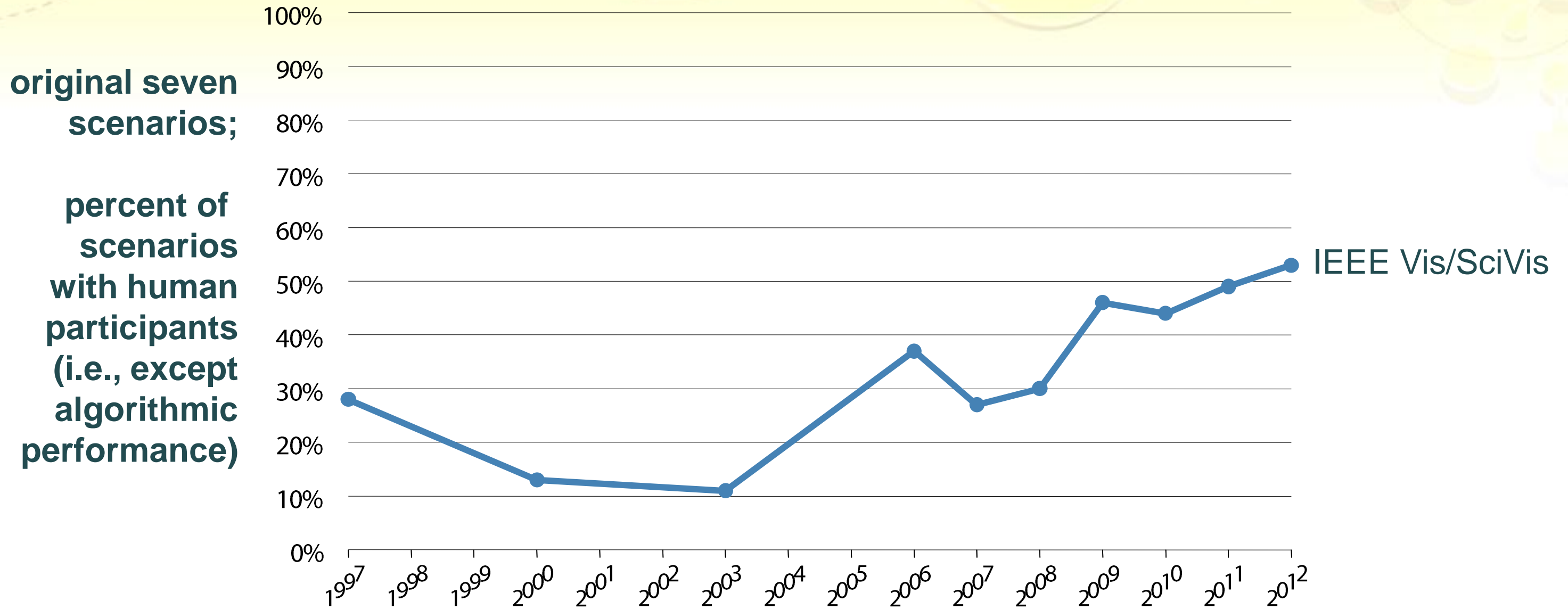


# Results: human participants



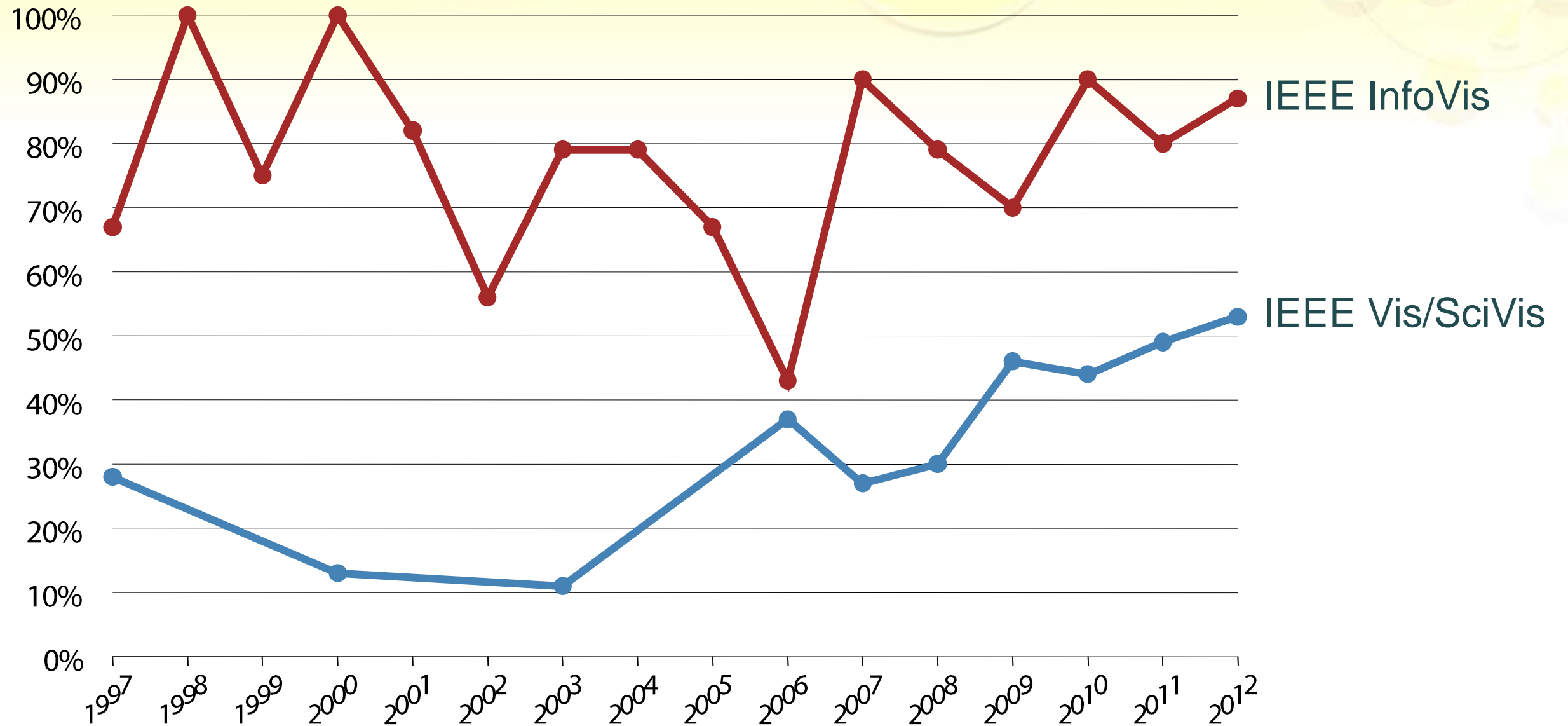


# Results: human participants

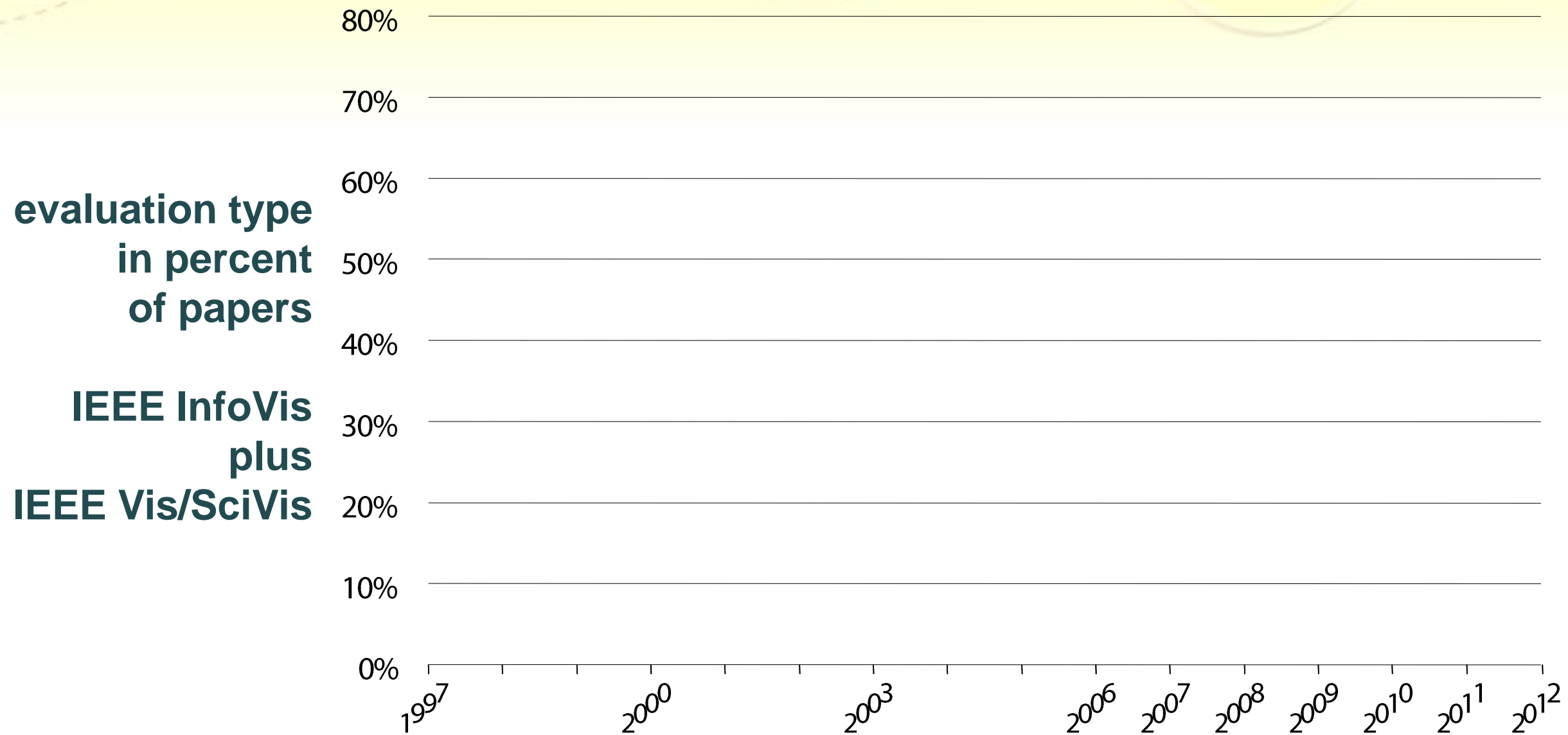


# Results: human participants

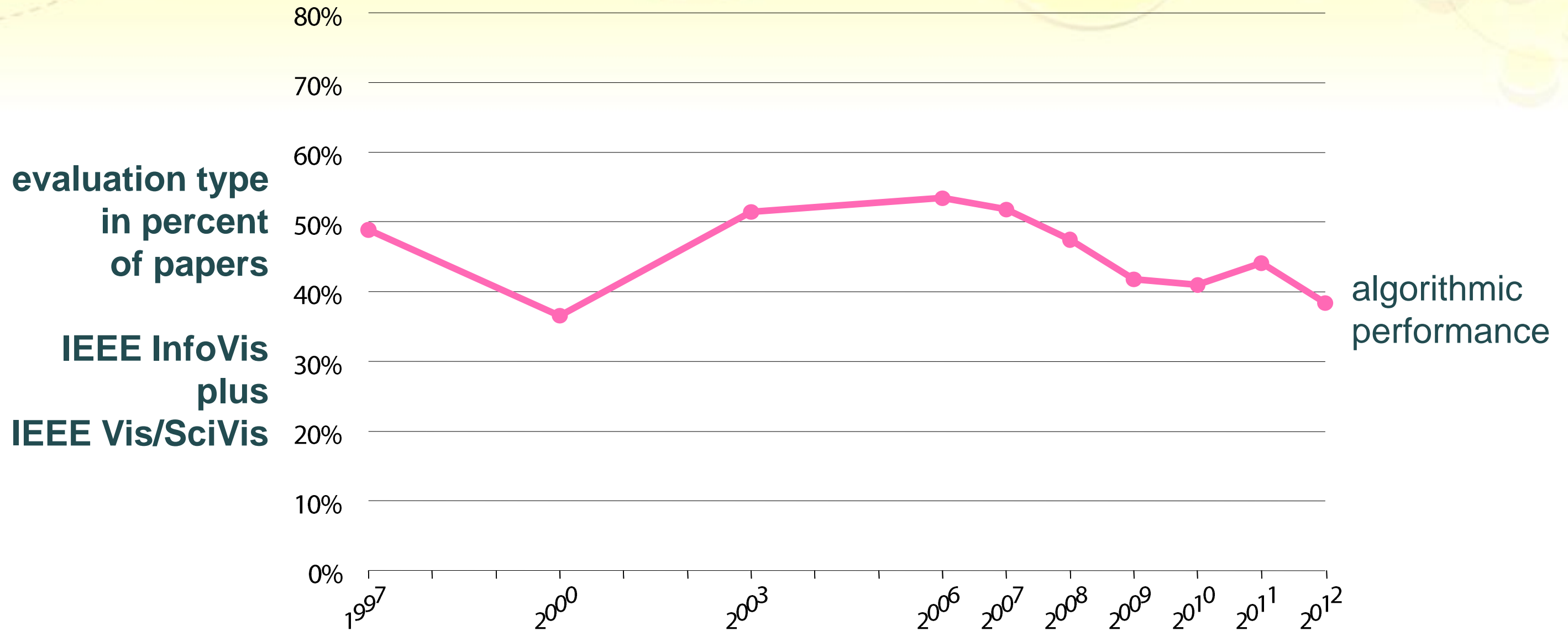
original seven scenarios;  
percent of scenarios with human participants (i.e., except algorithmic performance)



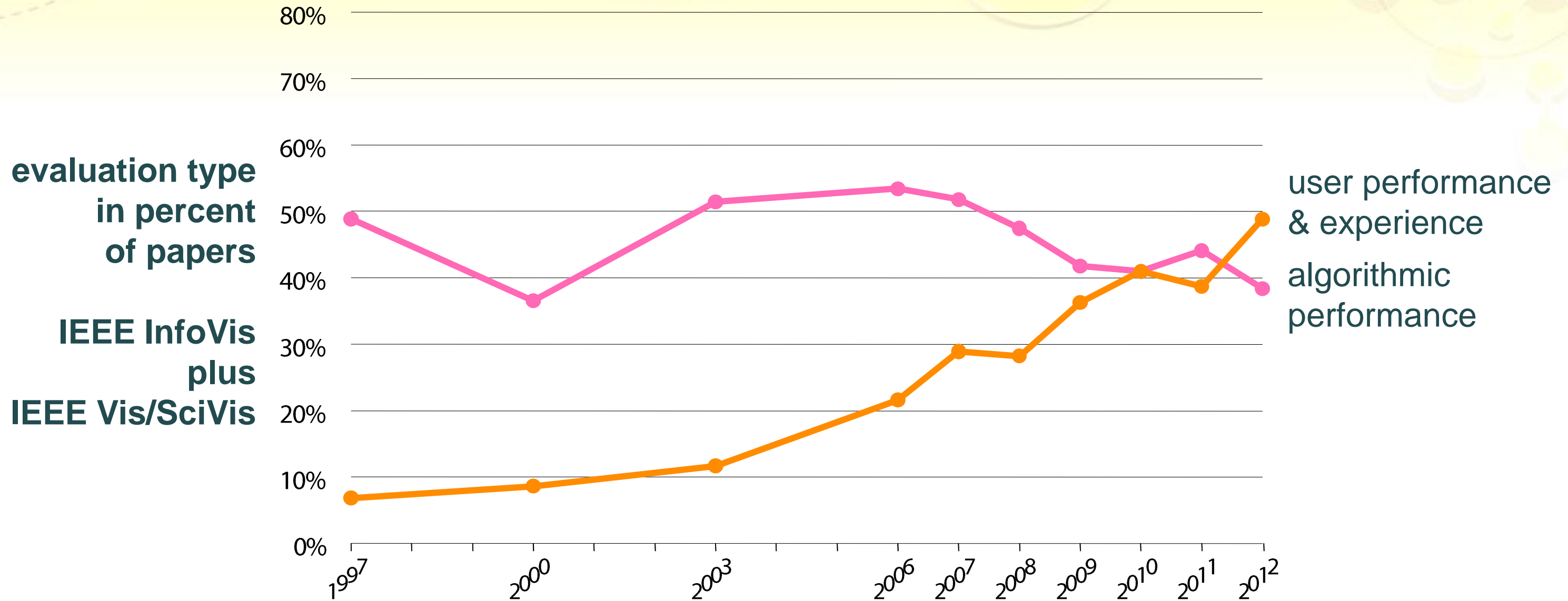
# Results: visualization community as a whole



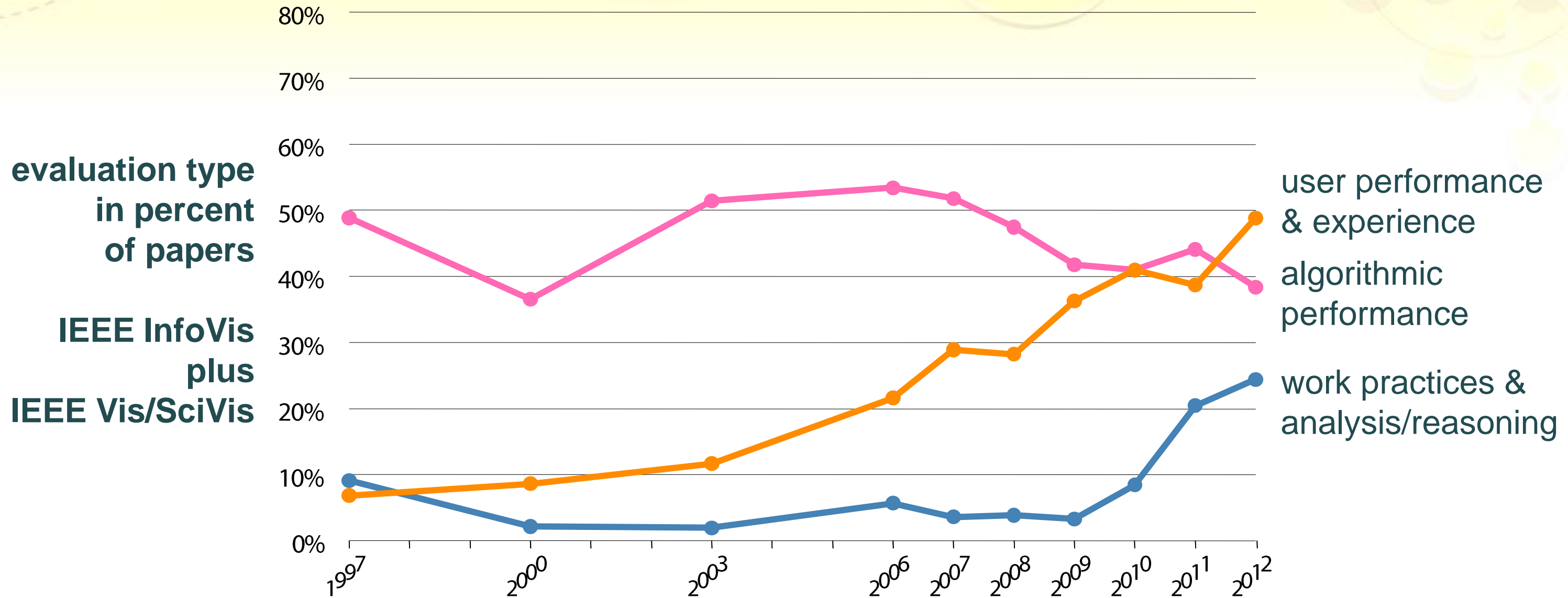
# Results: visualization community as a whole



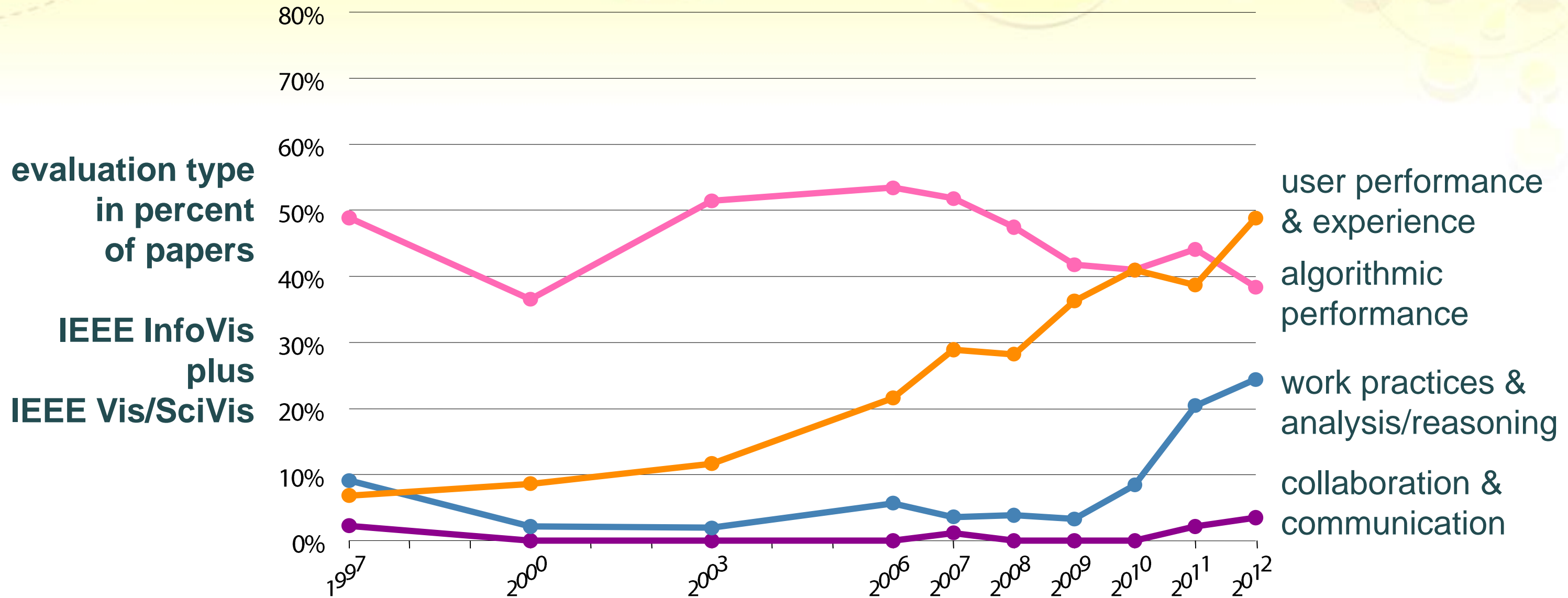
# Results: visualization community as a whole



# Results: visualization community as a whole



# Results: visualization community as a whole



# Considerations

- evaluation reporting rigor
- analyzing and reporting real problems
- statistical significance vs. qualitative expert feedback
- obtaining and reporting expert feedback
- use of case studies
- number of study participants

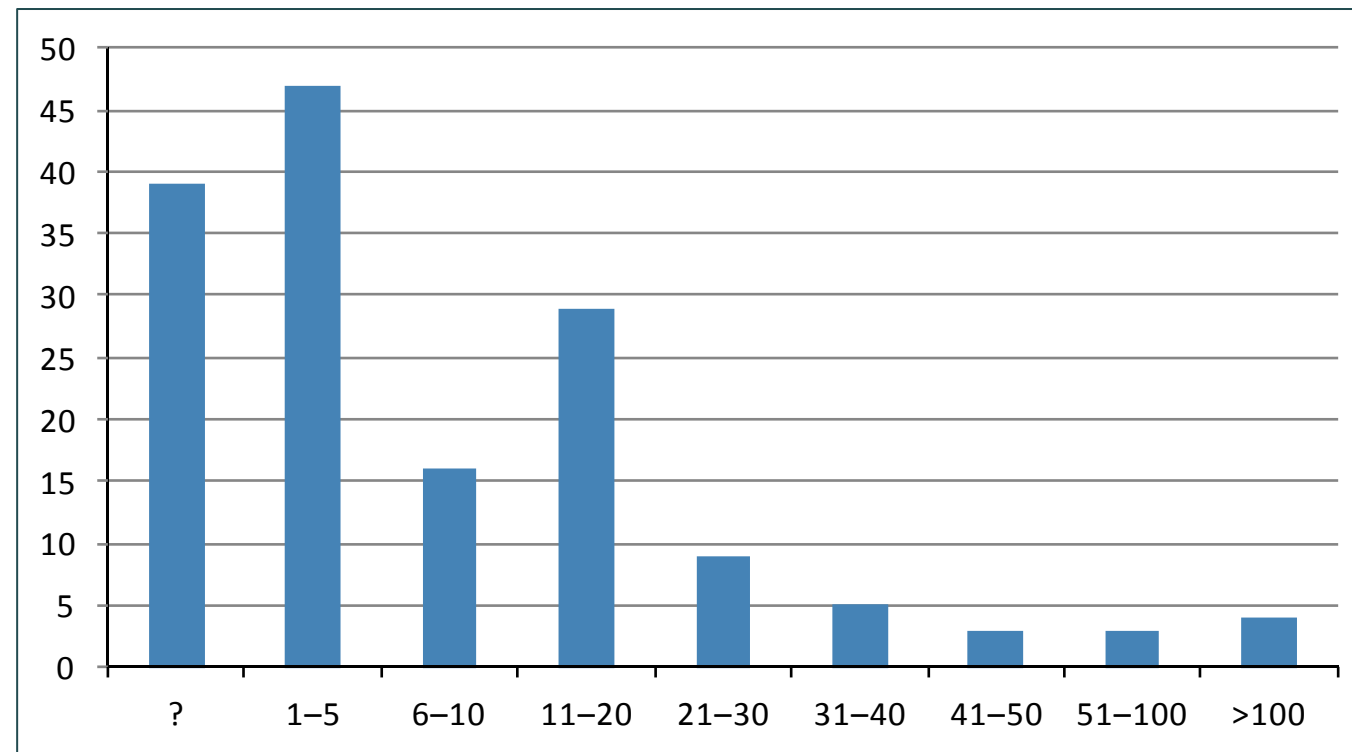


# Considerations

- evaluation reporting rigor
- analyzing and reporting real problems
- statistical significance vs. qualitative expert feedback
- obtaining and reporting expert feedback
- use of case studies
- number of study participants

# Evaluation reporting rigor

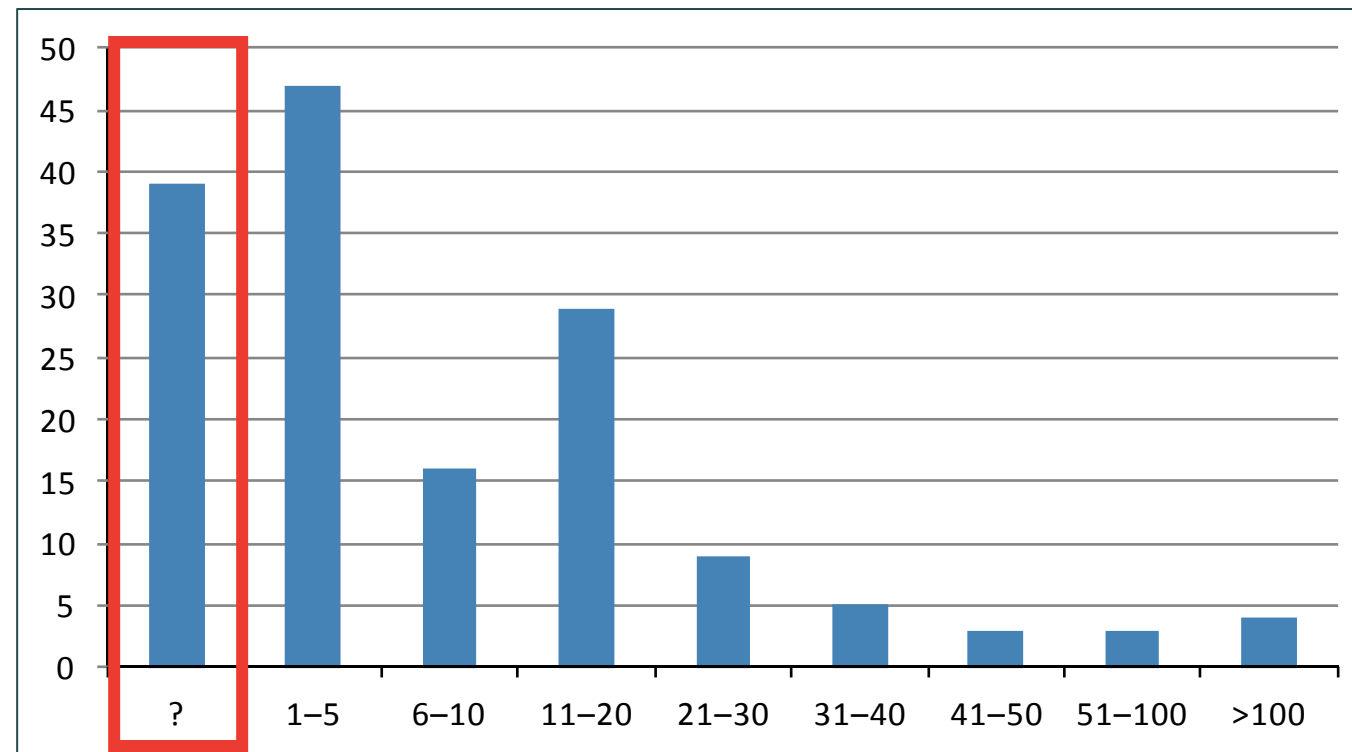
- too often detail missing about the evaluation
- need to report
  - participant details
  - collaboration details
  - evaluation protocols
  - ...



**number of  
participants  
per study**

# Evaluation reporting rigor

- too often detail missing about the evaluation
- need to report
  - participant details
  - collaboration details
  - evaluation protocols
  - ...

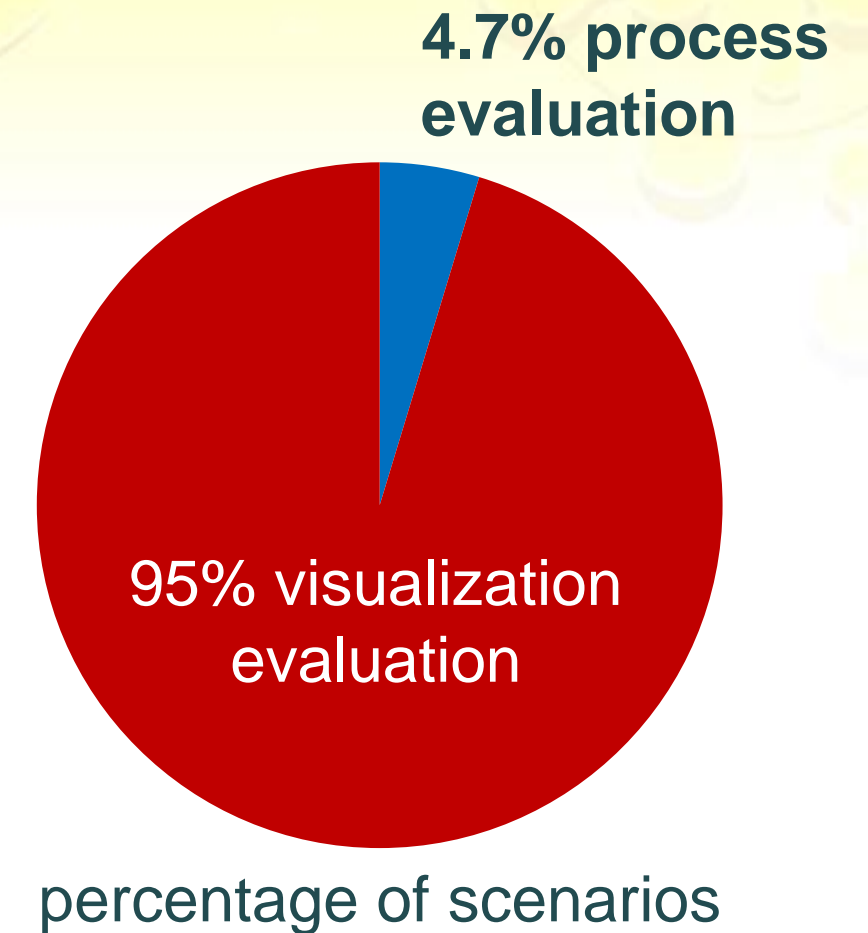


number of participants per study

>25% unclear

# Analyzing & reporting real problems

- relevant to virtually all visualization work:
  - understand visualization needs
  - understand use of visualizations for visual reasoning, communication, and collaboration
  - grounding work in reality
- in practice: often done!
  - describe work with experts
  - make process evaluations first-class citizens in our papers



# Statistical significance vs. expert feedback

- right methodology for given question!
- visualization: ill-defined, fuzzy, broad domain problems
- evaluation **not only** null hypothesis significance testing (NHST)

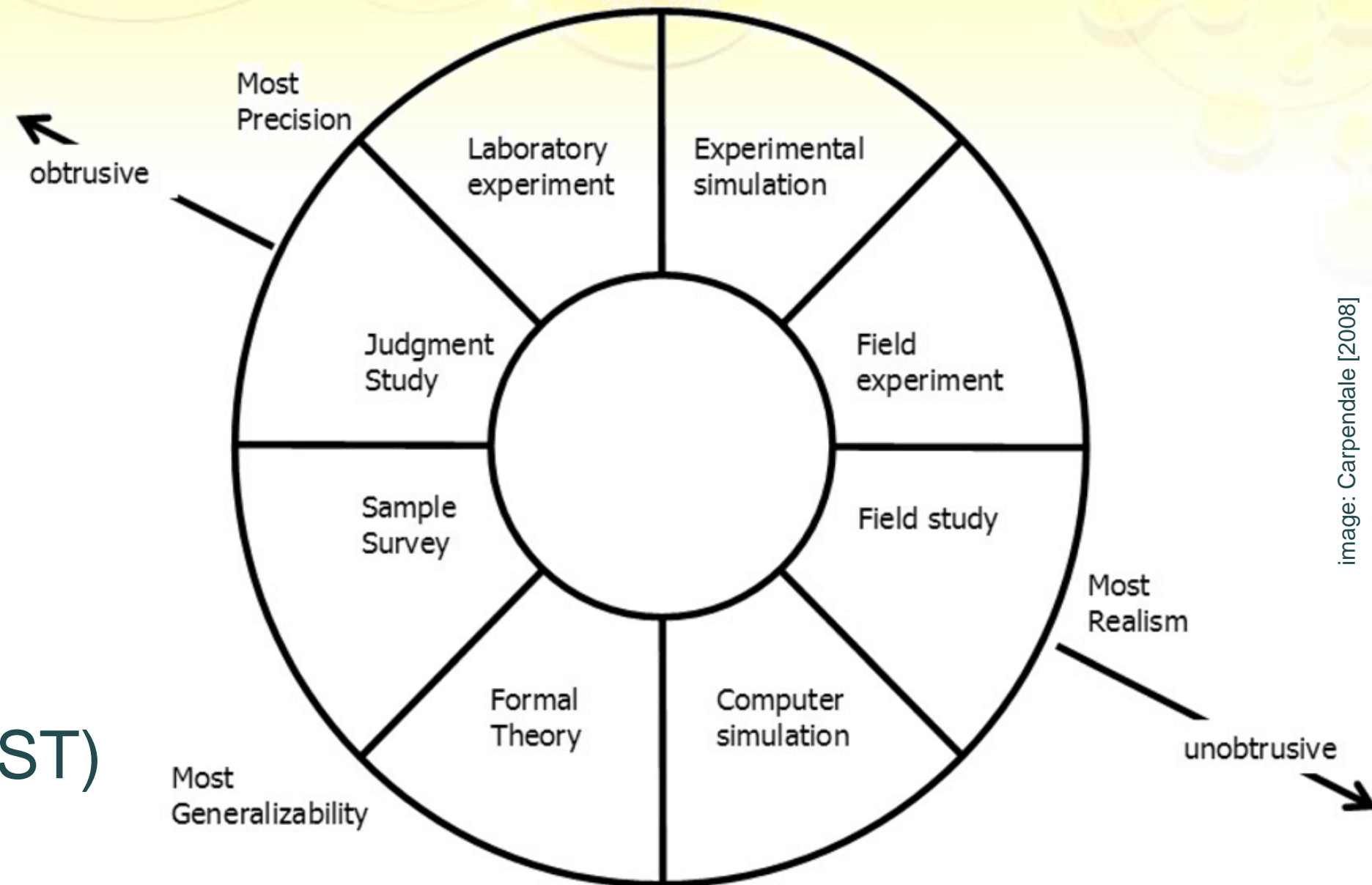


image: Carpendale [2008]

# Statistical significance vs. expert feedback

- right methodology for given question!
- visualization: ill-defined, fuzzy, broad domain problems
- evaluation **not only** null hypothesis significance testing (NHST)

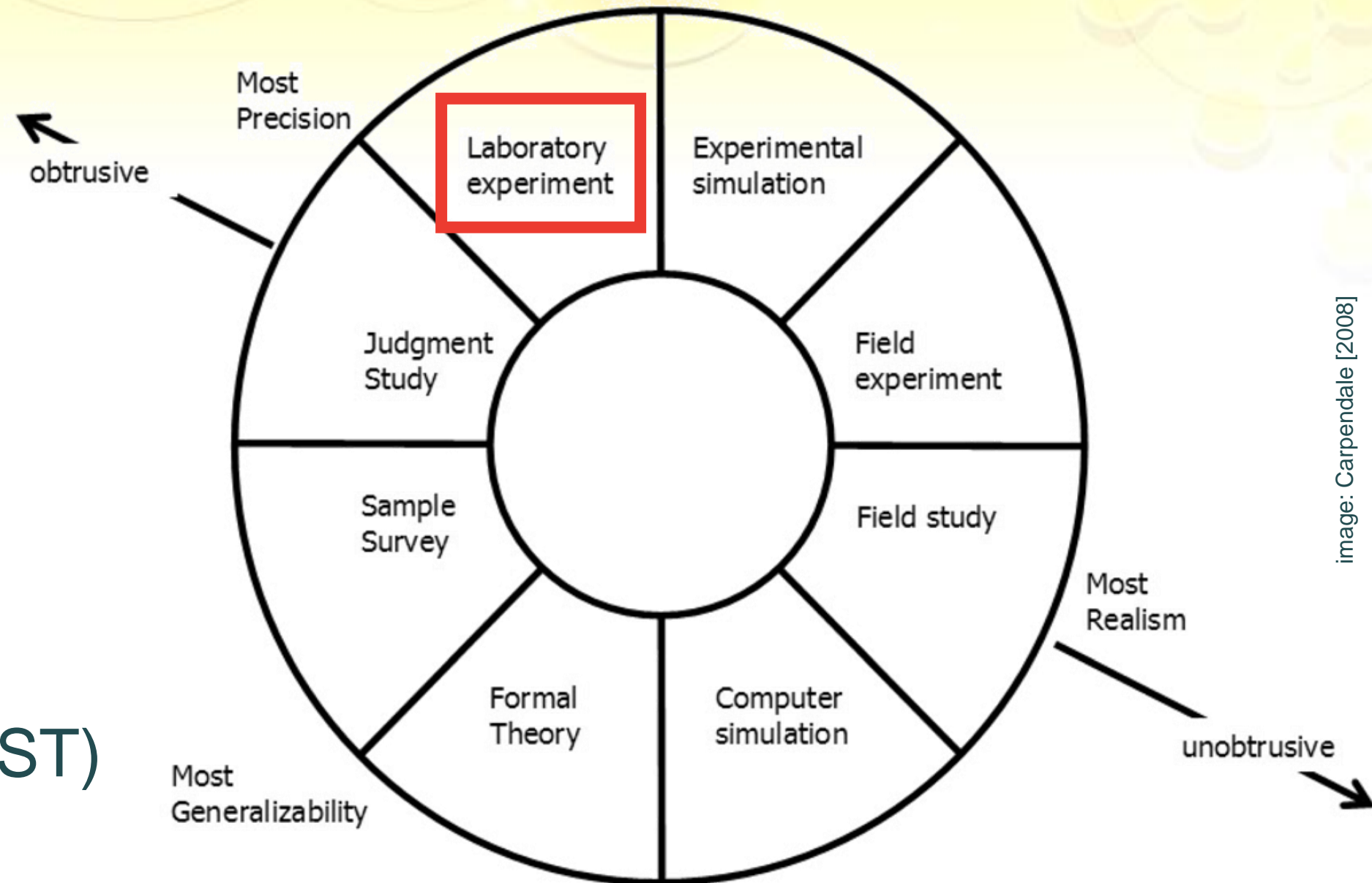


image: Carpendale [2008]

# Statistical significance vs. expert feedback

- right methodology for given question!
- visualization: ill-defined, fuzzy, broad domain problems
- evaluation **not only** null hypothesis significance testing (NHST)

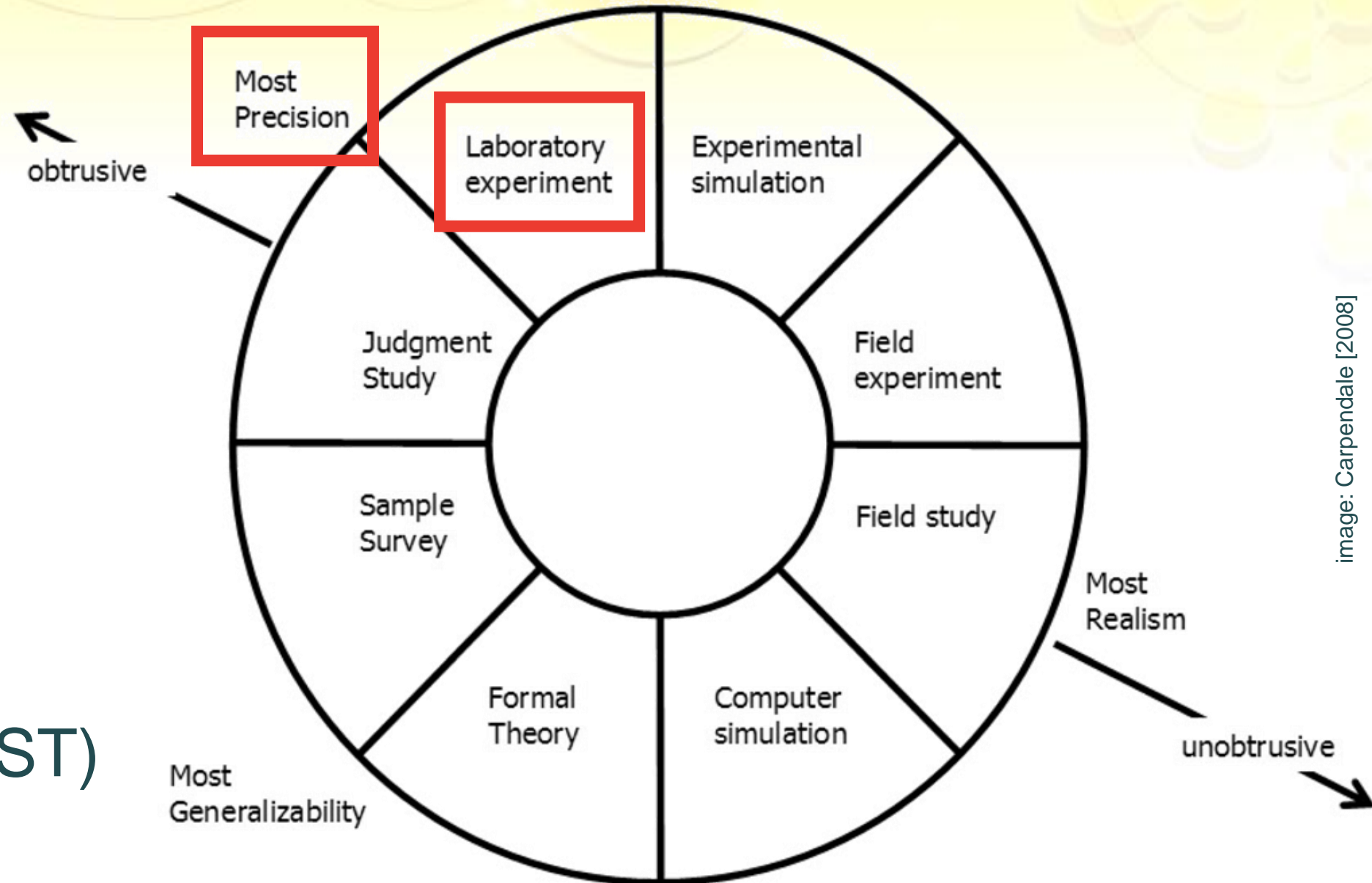


image: Carpendale [2008]

# Statistical significance vs. expert feedback

- right methodology for given question!
- visualization: ill-defined, fuzzy, broad domain problems
- evaluation **not only** null hypothesis significance testing (NHST)

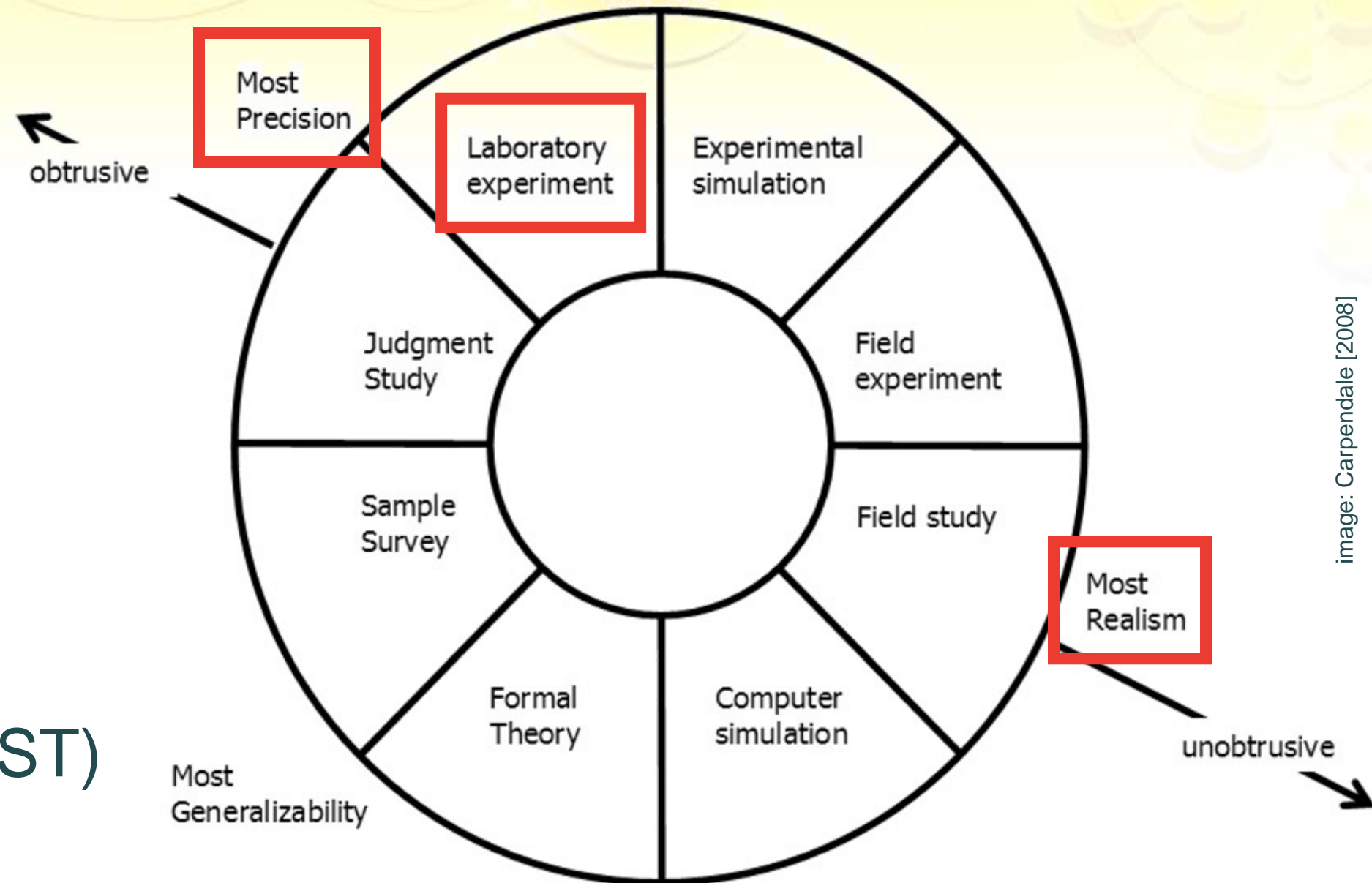
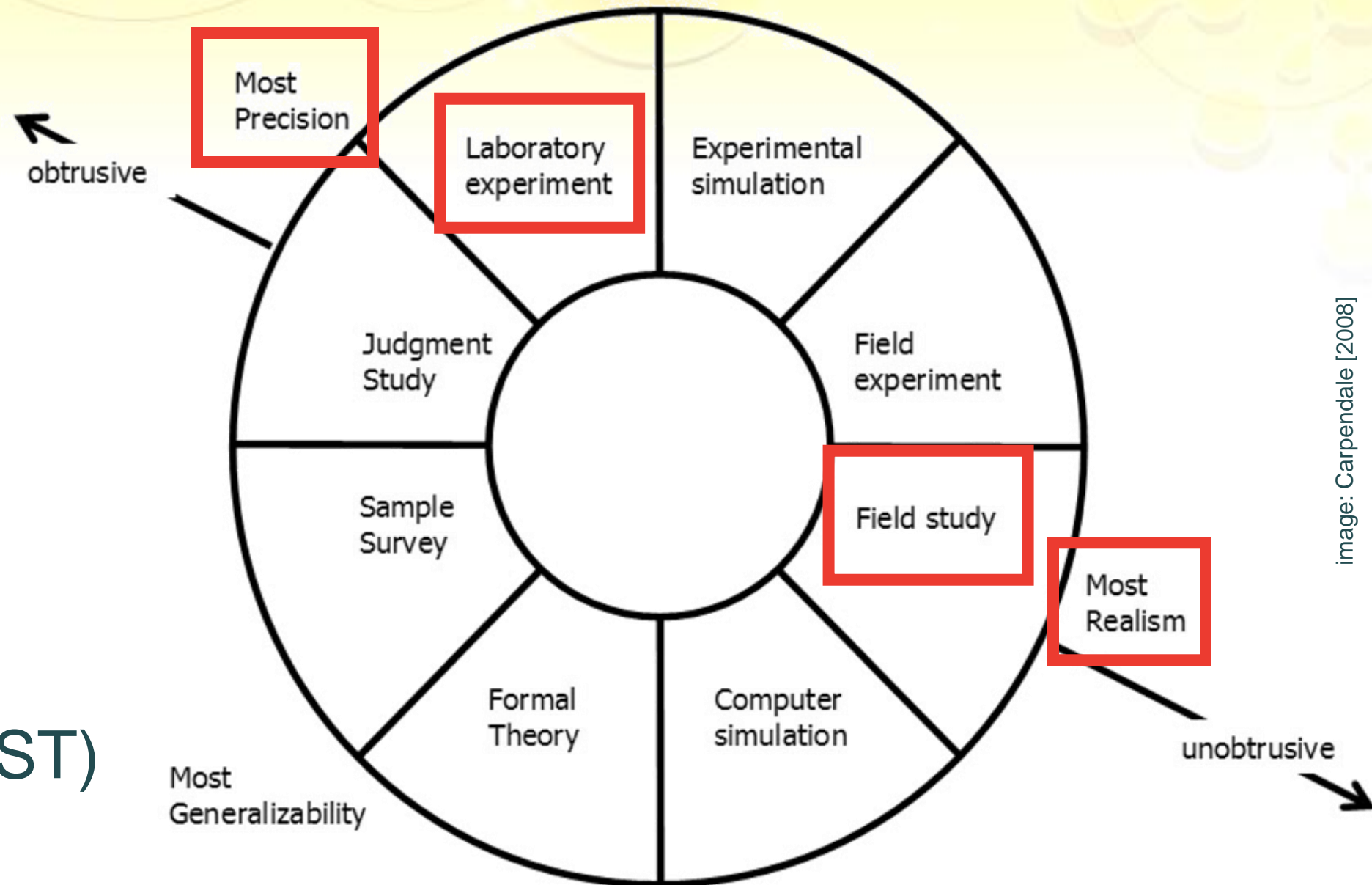


image: Carpendale [2008]



# Statistical significance vs. expert feedback

- right methodology for given question!
- visualization: ill-defined, fuzzy, broad domain problems
- evaluation **not only** null hypothesis significance testing (NHST)



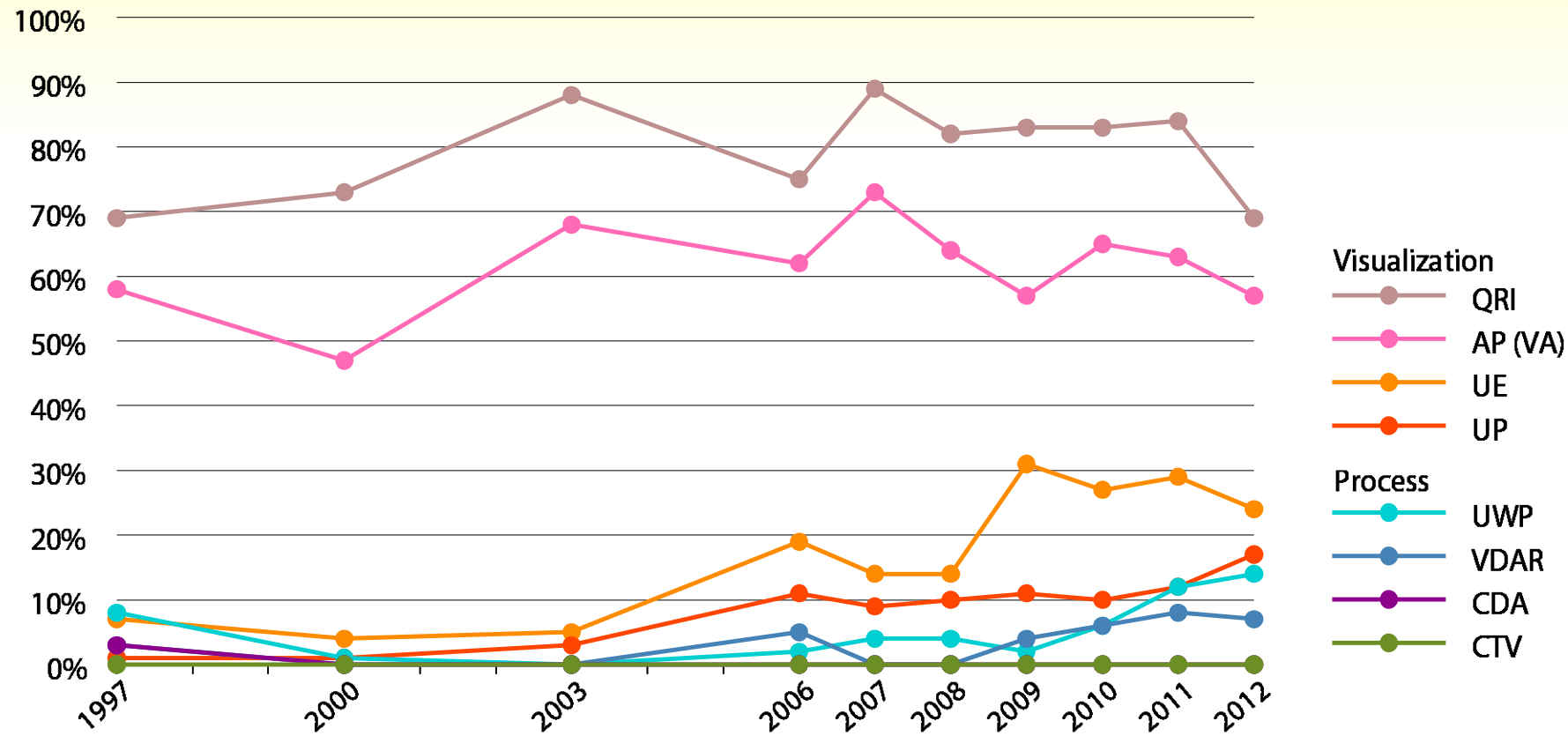
# Obtaining and reporting expert feedback

- *“We showed our system/tool to our collaborating experts and they really liked it.”*
- expert feedback valid & important
- but: **rigor** in study design and reporting!
- several guidelines on qualitative evaluation methods

# Open questions

- rigor in algorithmic performance?
  - how many datasets?
  - benchmark datasets?
- statistical analysis?
  - issues with NHST (see “dance of the p-values”)
  - how many participants?
- rigorous qualitative results inspection?

# Thanks for your attention



*shameless plug:*  
interested in working  
with us at *inria* informatics mathematics ?  
... talk to me

paper: <http://goo.gl/6yiggh>  
data: <http://goo.gl/CGswy>

