UNIVERSITY INSTITUTE OF LISBON

UNIVERSIDADE SALVADOR | UNIFACS LAUREATE INTERNATIONAL UNIVERSITIES

Survey on Code Smells

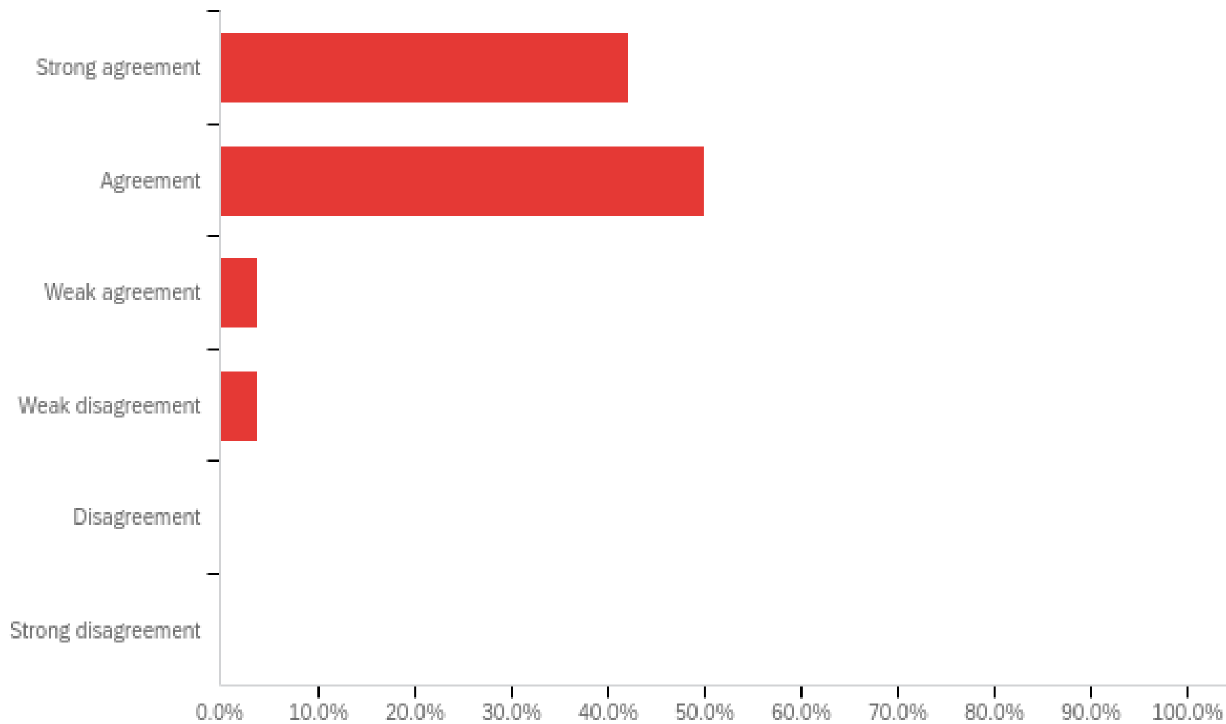
Authors – Descriptive statistics of responses

Rating scale used

Answer	Rating value
Strong agreement	1
Agreement	2
Weak agreement	3
Weak disagreement	4
Disagreement	5
Strong disagreement	6

Part 3 - Code smells detection techniques

Q3.2 - SLR FINDING: The most frequently used code smells detection techniques are based on rule-based approaches.



#	Field	Minimum	Maximum	Mean	Std Deviation	Variance	Count
1	SLR FINDING: The most frequently used code smells detection techniques are based on rule-based approaches.	1.0	4.0	1.7	0.7	0.5	26

#	SLR FINDING: The most frequently used code smells detection techniques are based on rule-based approaches.	Percentage
1	Strong agreement	42.3%
2	Agreement	50.0%
3	Weak agreement	3.8%
4	Weak disagreement	3.8%
5	Disagreement	0.0%

6	Strong disagreement	0.0%
	Total	26

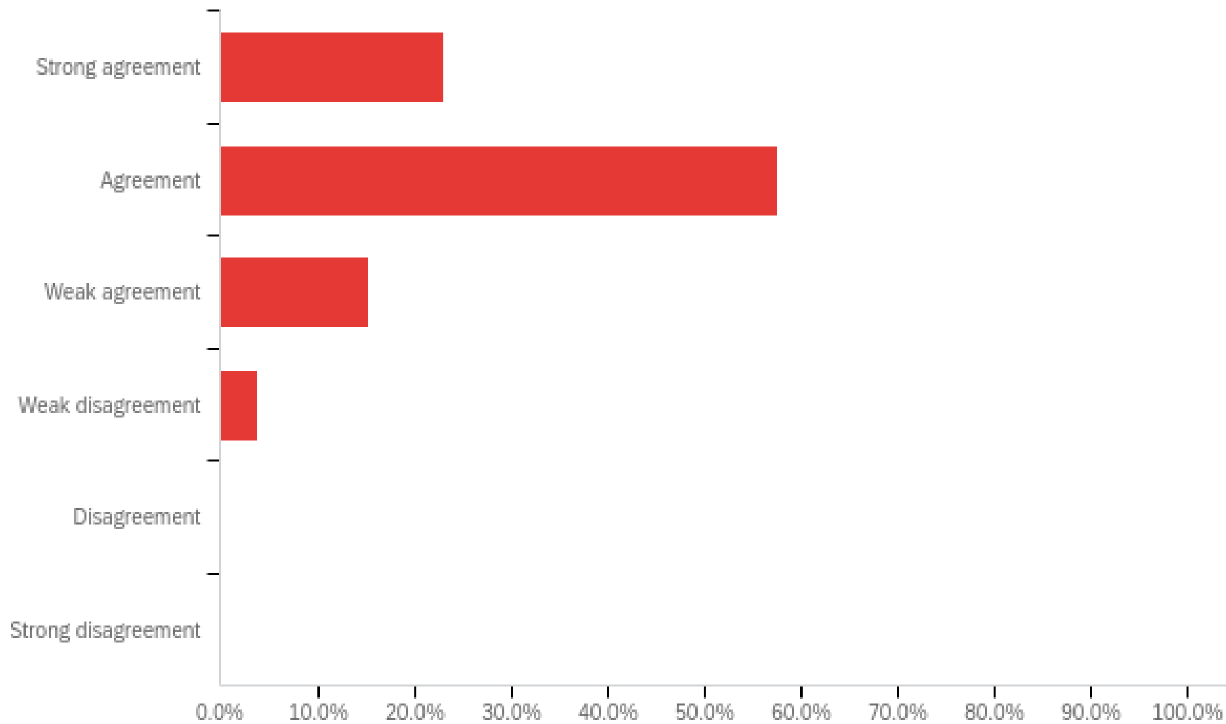
Q3.3 - How do you rate your confidence degree while assessing the previous finding?

#	Field	Minimum	Maximum	Mean	Std Deviation	Variance	Count
1	How do you rate your confidence degree while assessing the previous finding?	2.8	4.0	3.5	0.4	0.2	23

Q3.4 - Optional justification or comments

Optional justification or comments
What other kinds of detection techniques should exist? All of them are based on some kind of rule. I am not sure what specifically is meant by "rule-based".
We have used Machine Learning based approaches
the rule-based tools are dominating. Other approaches (structure-based, ML-based) are only emerging.

Q3.5 - SLR FINDING: Very few code smells detection studies provide their oracles (a tagged dataset for training detection algorithms).



#	Field	Minimum	Maximum	Mean	Std Deviation	Variance	Count
1	SLR FINDING: Very few code smells detection studies provide their oracles (a tagged dataset for training detection algorithms).	1.0	4.0	2.0	0.7	0.5	26

#	SLR FINDING: Very few code smells detection studies provide their oracles (a tagged dataset for training detection algorithms).	Percentage
1	Strong agreement	23.1%
2	Agreement	57.7%
3	Weak agreement	15.4%
4	Weak disagreement	3.8%
5	Disagreement	0.0%
6	Strong disagreement	0.0%
	Total	26

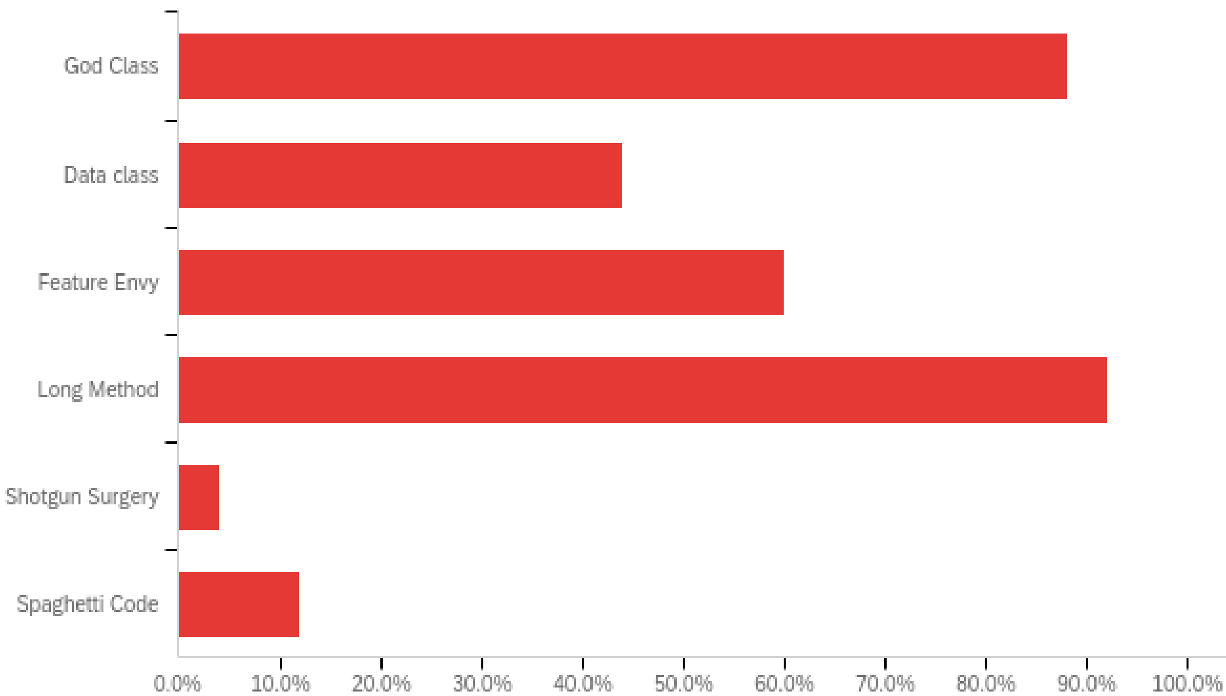
Q3.6 - How do you rate your confidence degree while assessing the previous finding?

#	Field	Minimum	Maximum	Mean	Std Deviation	Variance	Count
1	How do you rate your confidence degree while assessing the previous finding?	1.2	4.0	3.1	0.7	0.5	22

Q3.7 - Optional justification or comments

Optional justification or comments
Most of the studies provide dataset
This is not surprising, code smells are inherently subjective. It would be difficult to point to a definitely smelly class.
Only some recent studies have made their datasets publicly available

Q3.8 - OPINION: Please select the 3 most often detected code smells.



#	OPINION: Please select the 3 most often detected code smells.	Percentage
1	God Class	29.3%
2	Data class	14.7%
3	Feature Envy	20.0%
4	Long Method	30.7%
5	Shotgun Surgery	1.3%
6	Spaghetti Code	4.0%
	Total	75

Q3.9 - How do you rate your confidence degree while expressing the previous opinion?

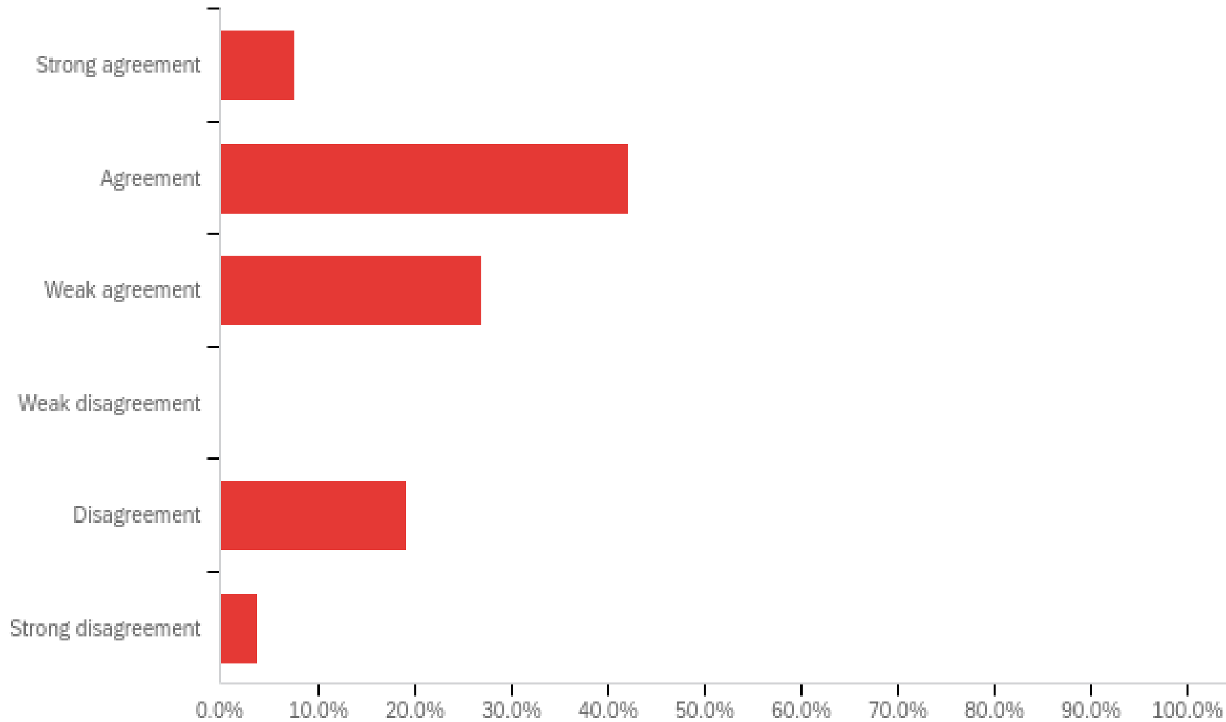
#	Field	Minimum	Maximum	Mean	Std Deviation	Variance	Count
1	How do you rate your confidence degree while expressing the previous opinion?	2.0	4.0	3.3	0.5	0.3	22

Q3.10 - Optional justification or comments

Optional justification or comments
I believe "Duplicated Code" is the most frequently detected kind of bad smell, but it is not offered in the list above.
they are easily detectable using rules, so most of tools focus on them.

Part 4 - Code smells detection effectiveness

Q4.1 - SLR FINDING: In the detection of simpler code smells (e.g. Long Method or God Class), the achieved precision and recall of detection techniques can be very high (up to 100%).



#	Field	Minimum	Maximum	Mean	Std Deviation	Variance	Count
1	SLR FINDING: In the detection of simpler code smells (e.g. Long Method or God Class), the achieved precision and recall of detection techniques can be very high (up to 100%).	1.0	6.0	2.9	1.4	1.8	26

#	SLR FINDING: In the detection of simpler code smells (e.g. Long Method or God Class), the achieved precision and recall of detection techniques can be very high (up to 100%).	Percentage
1	Strong agreement	7.7%
2	Agreement	42.3%
3	Weak agreement	26.9%

4	Weak disagreement	0.0%
5	Disagreement	19.2%
6	Strong disagreement	3.8%
	Total	26

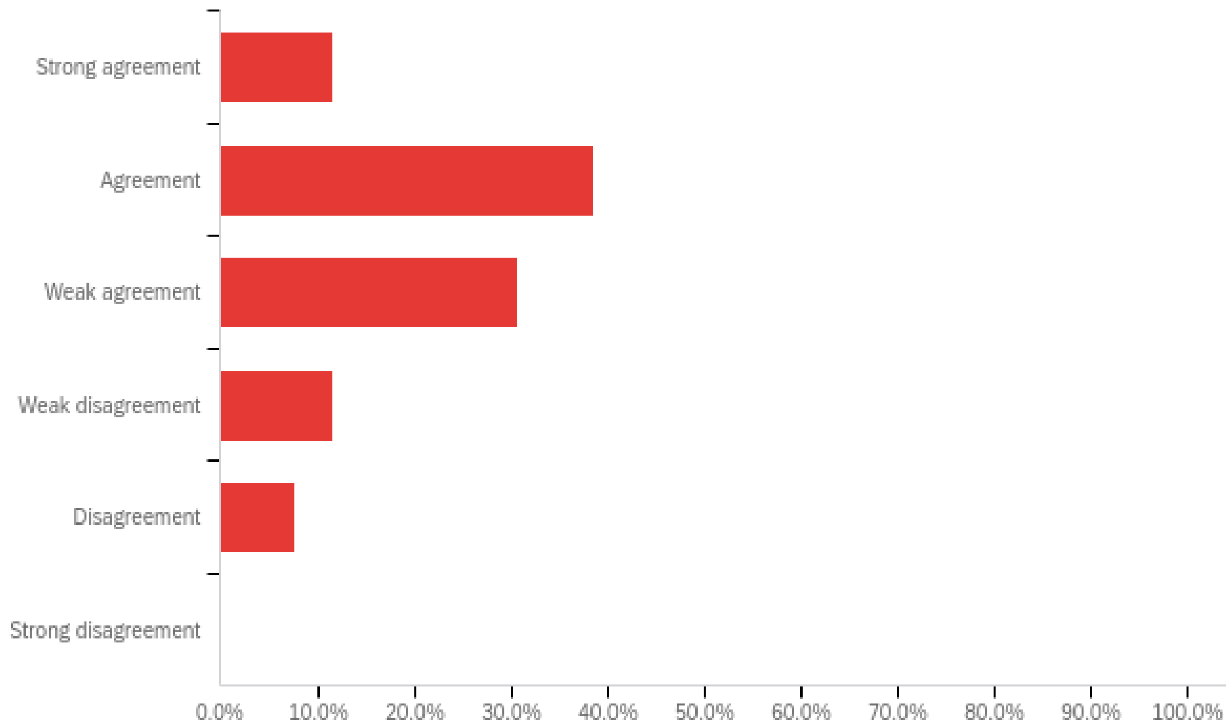
Q4.2 - How do you rate your confidence degree while assessing the previous finding?

#	Field	Minimum	Maximum	Mean	Std Deviation	Variance	Count
1	How do you rate your confidence degree while assessing the previous finding?	2.3	4.0	3.2	0.5	0.3	20

Q4.3 - Optional justification or comments

Optional justification or comments
Precision and recall is difficult to measure for code smells, as it often depends on contextual factors.
I am very confident w.r.t. Long Method because it is very simply to detect if one agrees on the threshold for "long". God Class is more difficult to define (multiple combined criteria), hence, agreement on the criteria is generally lower.
The detection of these smells is very sensitive to the developer’s perception. Thus, a method can be long for a developer but not for another developer.
it depends also on the size of the system and sometimes on the system domain
They CAN be very high, but it strongly depends on context (in particular, for multi-symptoms smells, like God Class)
It depends on how the dataset is built. In our research, we showed that structural-based detectors are not able to identify all the instances of a certain code smell. So, I would say that 100% precision and recall can be only achieved on a very few datasets, but the result cannot be generalizable.

Q4.4 - SLR FINDING: When the complexity of code smells is greater (e.g. Divergent Change or Shotgun Surgery), the precision and recall in detection are much lower than in simpler code smells.



#	Field	Minimum	Maximum	Mean	Std Deviation	Variance	Count
1	SLR FINDING: When the complexity of code smells is greater (e.g. Divergent Change or Shotgun Surgery), the precision and recall in detection are much lower than in simpler code smells.	1.0	5.0	2.7	1.1	1.1	26

#	SLR FINDING: When the complexity of code smells is greater (e.g. Divergent Change or Shotgun Surgery), the precision and recall in detection are much lower than in simpler code smells.	Percentage
1	Strong agreement	11.5%
2	Agreement	38.5%
3	Weak agreement	30.8%
4	Weak disagreement	11.5%
5	Disagreement	7.7%

6	Strong disagreement	0.0%
	Total	26

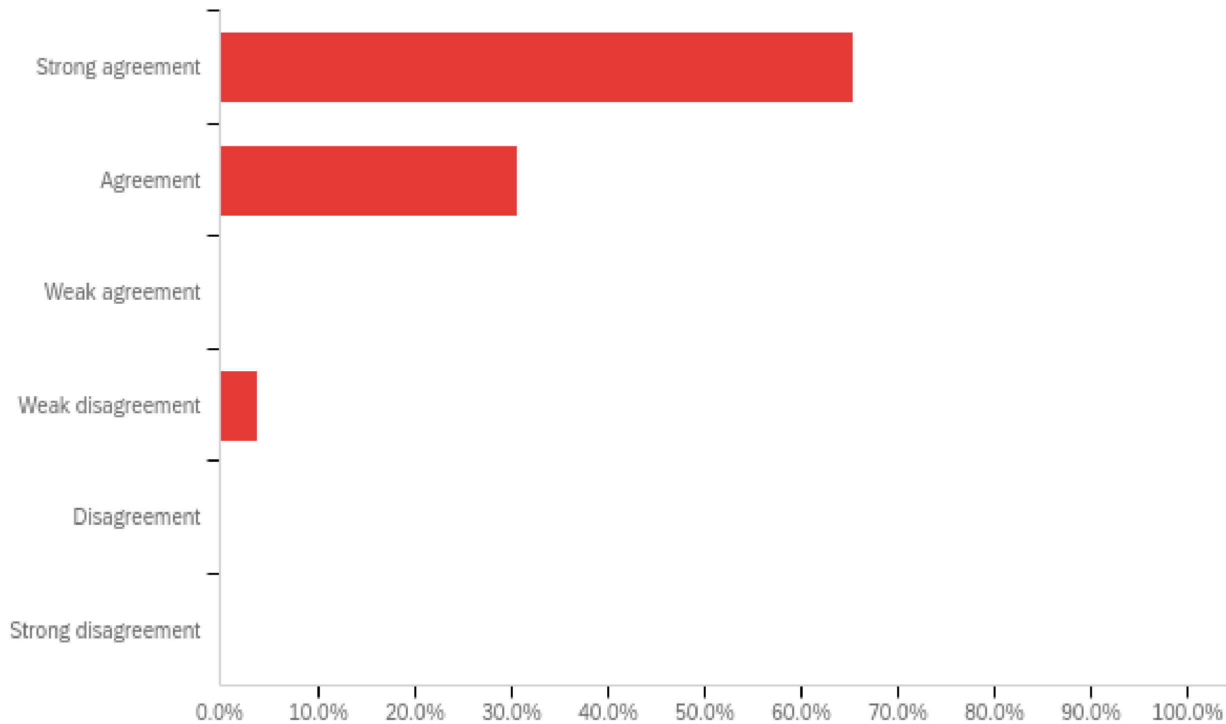
Q4.5 - How do you rate your confidence degree while assessing the previous finding?

#	Field	Minimum	Maximum	Mean	Std Deviation	Variance	Count
1	How do you rate your confidence degree while assessing the previous finding?	2.6	4.0	3.2	0.4	0.2	17

Q4.6 - Optional justification or comments

Optional justification or comments
Not completely sure on what do you mean by "complexity of code smells". You mean the properties of these are more difficult to specify?
See my comment in previous question
It could be also higher
Not a necessary correspondence - a bottom-to-top incremental detector might have high confidence also for complex code smells if their constituting simpler indicator smells are detected with high confidence.
Unsure.
I cannot understand what complexity means in this case, however I would say that there are NOT simpler or more complex smells, but only approaches that can detect them better or not on the basis of their peculiar characteristics (e.g., A shotgun surgery cannot properly be detected via source code analysis but it requires some historical analysis).

Q4.7 - SLR FINDING: There are few oracles (a tagged dataset for training detection algorithms) shared and publicly available. The existence of shared and collaborative oracles could improve the state of the art in code smells detection research.



#	Field	Minimum	Maximum	Mean	Std Deviation	Variance	Count
1	SLR FINDING: There are few oracles (a tagged dataset for training detection algorithms) shared and publicly available. The existence of shared and collaborative oracles could improve the state of the art in code smells detection research.	1.0	4.0	1.4	0.7	0.5	26

#	SLR FINDING: There are few oracles (a tagged dataset for training detection algorithms) shared and publicly available. The existence of shared and collaborative oracles could improve the state of the art in code smells detection research.	Percentage
1	Strong agreement	65.4%
2	Agreement	30.8%
3	Weak agreement	0.0%
4	Weak disagreement	3.8%

5	Disagreement	0.0%
6	Strong disagreement	0.0%
	Total	26

Q4.8 - How do you rate your confidence degree while assessing the previous finding?

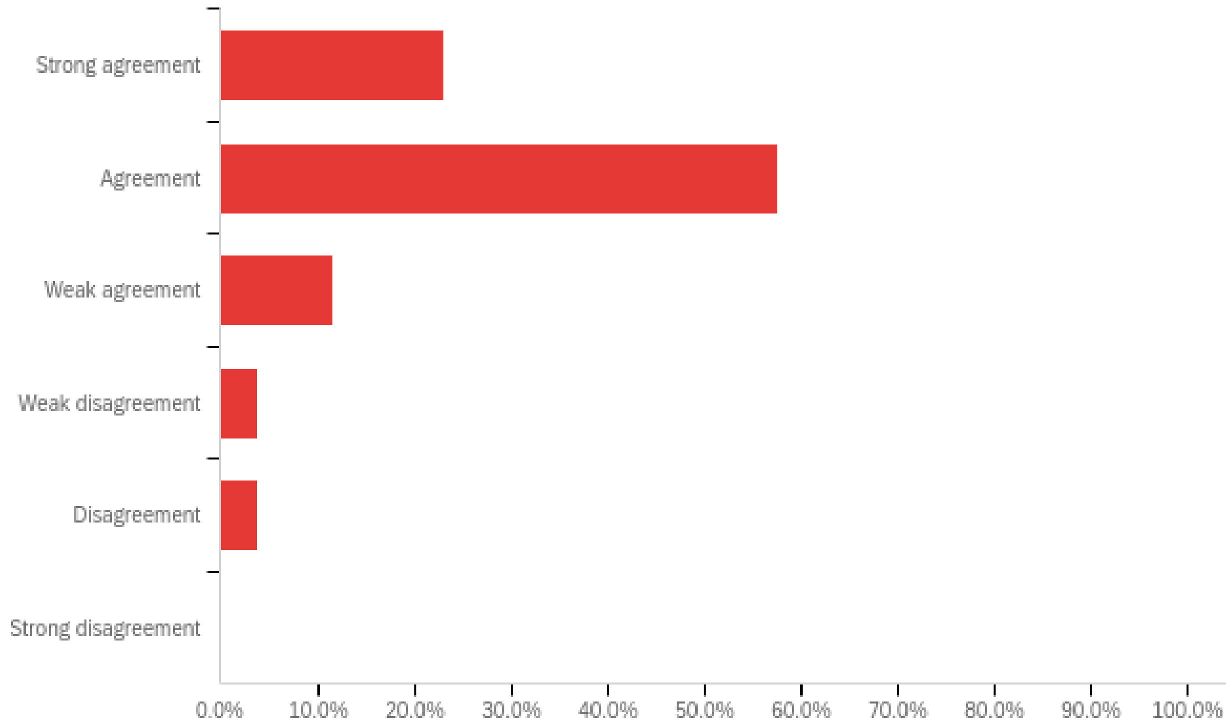
#	Field	Minimum	Maximum	Mean	Std Deviation	Variance	Count
1	How do you rate your confidence degree while assessing the previous finding?	2.4	4.0	3.6	0.5	0.3	21

Q4.9 - Optional justification or comments

Optional justification or comments
Oracles are also important for replication and reproduction of empirical studies.
I generally agree. But benchmarks have the risks that everyone tries to optimize their technique regarding to the existing benchmarks -- something we have seen, for instance, in performance benchmarks for processors (CPUs).
http://soft.vub.ac.be/landfill/
Note that you are involving two affirmations.
Definitely that would allow for comparison of results and identification of contextual factors in smell detection.

Part 2 - Code smells visualization

Q2.2 - SLR FINDING: The vast majority of code smells detection studies do not propose visualization features for their detection.



#	Field	Minimum	Maximum	Mean	Std Deviation	Variance	Count
1	SLR FINDING: The vast majority of code smells detection studies do not propose visualization features for their detection.	1.0	5.0	2.1	0.9	0.8	26

#	SLR FINDING: The vast majority of code smells detection studies do not propose visualization features for their detection.	Percentage
1	Strong agreement	23.1%
2	Agreement	57.7%
3	Weak agreement	11.5%
4	Weak disagreement	3.8%

5	Disagreement	3.8%
6	Strong disagreement	0.0%
	Total	26

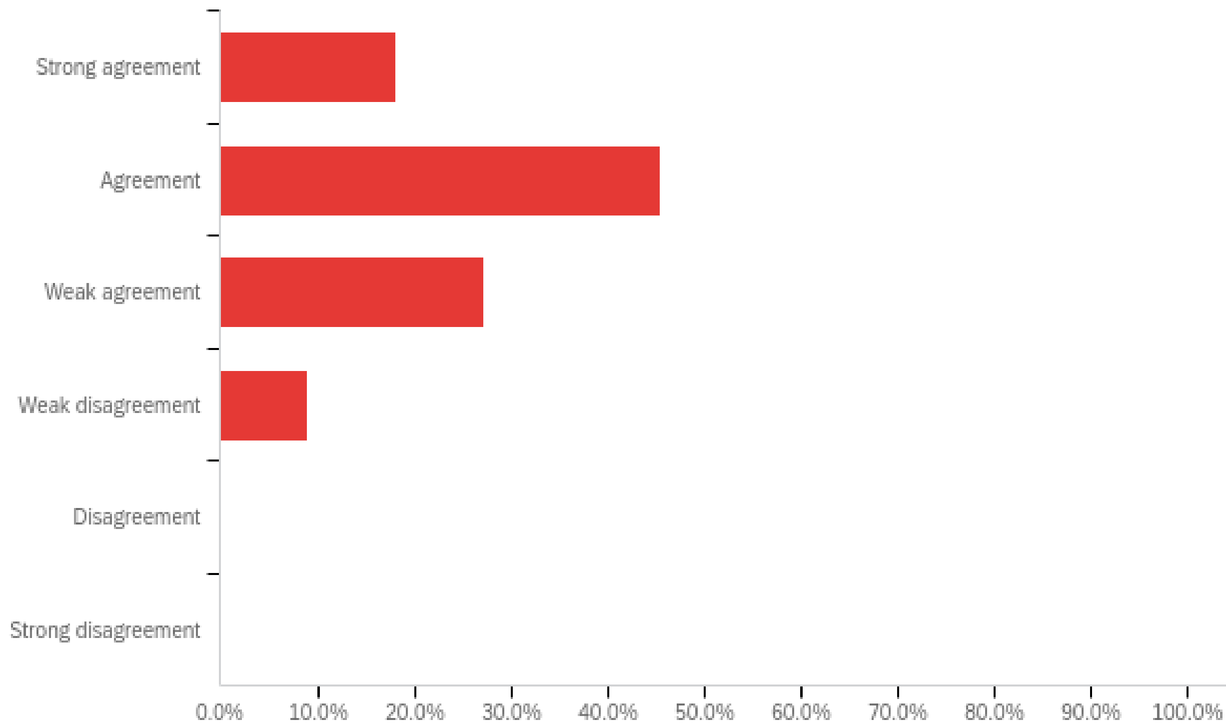
Q2.3 - How do you rate your confidence degree while assessing the previous finding?

#	Field	Minimum	Maximum	Mean	Std Deviation	Variance	Count
1	How do you rate your confidence degree while assessing the previous finding?	1.3	4.0	3.2	0.7	0.5	20

Q2.4 - Optional justification or comments

Optional justification or comments
Detection and visualization are separate issues. A paper should generally focus on only one of these aspects (separation of concerns holds true also for research papers). Papers are limited in space and generally you cannot do a good job in covering both aspects. For clone detection, there are quite a few visualization papers. Moreover, first things first. First we need to solve the problem of detection code smells, then only we can try to visualize them. Hence, it may not be surprising that we have fewer visualizations than detection techniques at this stage of research majurity.
It depends by what we mean by "visualization". I am not sure that developers would need such features (if they would, the features would be implemented).
I am not quite sure what you mean by visualization stickers that are few tools that gave examples of code snippets to explain the code smells and I'm not sure if that what you mean

Q2.5 - SLR FINDING: The vast majority of existing code smells visualization studies did not present evidence of its usage upon large software systems.



#	Field	Minimum	Maximum	Mean	Std Deviation	Variance	Count
1	SLR FINDING: The vast majority of existing code smells visualization studies did not present evidence of its usage upon large software systems.	1.0	4.0	2.3	0.9	0.7	22

#	SLR FINDING: The vast majority of existing code smells visualization studies did not present evidence of its usage upon large software systems.	Percentage
1	Strong agreement	18.2%
2	Agreement	45.5%
3	Weak agreement	27.3%
4	Weak disagreement	9.1%
5	Disagreement	0.0%
6	Strong disagreement	0.0%
	Total	22

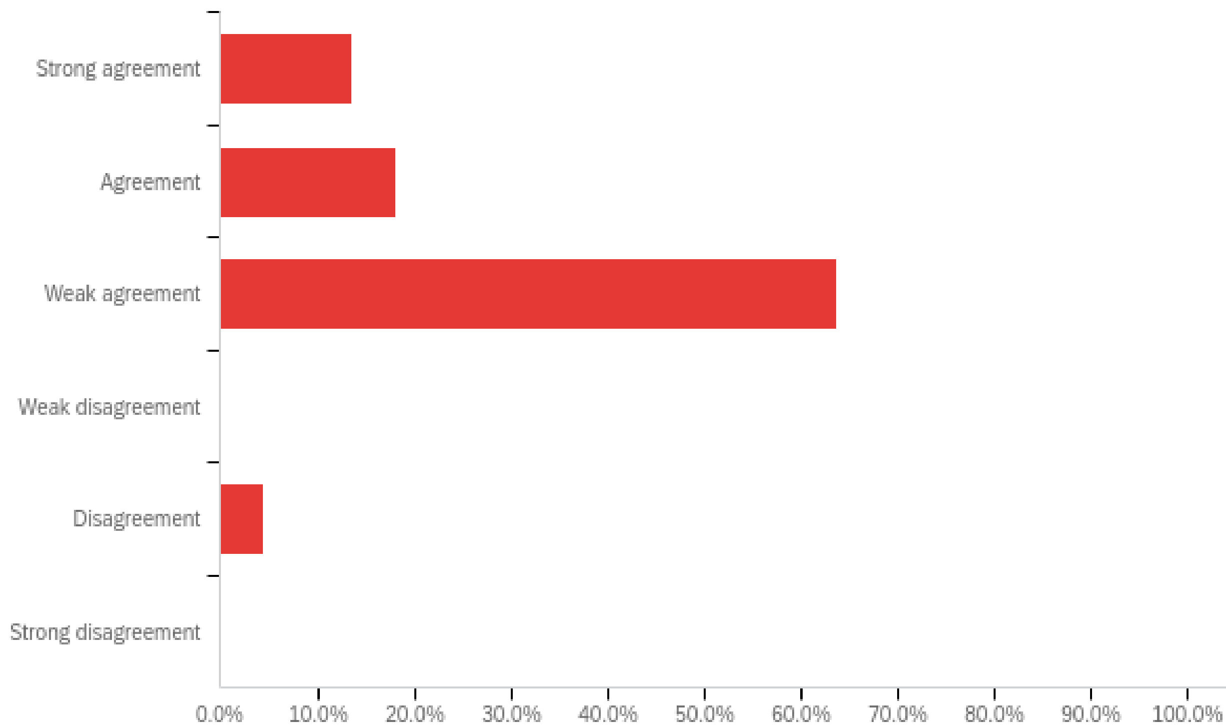
Q2.6 - How do you rate your confidence degree while assessing the previous finding?

#	Field	Minimum	Maximum	Mean	Std Deviation	Variance	Count
1	How do you rate your confidence degree while assessing the previous finding?	0.0	4.0	2.9	1.1	1.1	16

Q2.7 - Optional justification or comments

Optional justification or comments
It is difficult to properly assess a visualization approach.
Unfortunately, that is true for most visualization papers. It is certainly true for visualization of software clones. We are currently working on a systematic mapping study on clone visualization and have found hardly any kind of empirical evaluation of the proposed visualization techniques.
Do not know enough code smells visualization studies to make a statement on that.
To be honest, I don't know.
it would be better to have an option called I do not know because for this question for instance I don't have the right answer

Q2.8 - SLR FINDING: Software visualization researchers have not adopted specific visualization related taxonomies, such as the ones below, to support the identification of code smells: B. Price, R. Baecker, I. Small, A principled taxonomy of software visualization, Journal of Visual Languages and Computing 4 (3) (1993) 211–266. Roman, G. C., & Cox, K. C. (1993).A taxonomy of program visualization systems. Computer, 26(12), 11-24. Maletic, J. I., Marcus, A., & Collard, M. L. (2002). A task oriented view of software visualization. In Proceedings First International Workshop on Visualizing Software for Understanding and Analysis (pp.32-40). IEEE. Gallagher, K., Hatch, A., & Munro, M. (2008). Software architecture visualization: An evaluation framework and its application. IEEE Transactions on Software Engineering, 34(2), 260-270. Myller, N., Bednarik, R., Sutinen, E., & Ben-Ari, M. (2009). Extending the engagement taxonomy: Software visualization and collaborative learning. ACM Transactions on Computing Education (TOCE), 9(1), 7.



#	Field	Minimum	Maximum	Mean	Std Deviation	Variance	Count
1	SLR FINDING: Software visualization researchers have not adopted specific visualization related taxonomies, such as the ones below, to support the identification of code smells: B. Price, R. Baecker, I. Small, A principled taxonomy of software visualization, Journal of Visual	1.0	5.0	2.6	0.9	0.8	22

	Languages and Computing 4 (3) (1993) 211–266. Roman, G. C., & Cox, K. C. (1993).A taxonomy of program visualization systems. Computer, 26(12), 11-24. Maletic, J. I., Marcus, A., & Collard, M. L. (2002). A task oriented view of software visualization. In Proceedings First International Workshop on Visualizing Software for Understanding and Analysis (pp.32-40). IEEE. Gallagher, K., Hatch, A., & Munro, M. (2008). Software architecture visualization: An evaluation framework and its application. IEEE Transactions on Software Engineering, 34(2), 260-270. Myller, N., Bednarik, R., Sutinen, E., & Ben-Ari, M. (2009). Extending the engagement taxonomy: Software visualization and collaborative learning. ACM Transactions on Computing Education (TOCE), 9(1), 7.						

#	SLR FINDING: Software visualization researchers have not adopted specific visualization related taxonomies, such as the ones below, to support the identification of code smells: B. Price, R. Baecker, I. Small, A principled taxonomy of software visualization, Journal of Visual Languages and Computing 4 (3) (1993) 211–266. Roman, G. C., & Cox, K. C. (1993).A taxonomy of program visualization systems. Computer, 26(12), 11-24. Maletic, J. I., Marcus, A., & Collard, M. L. (2002). A task oriented view of software visualization. In Proceedings First International Workshop on Visualizing Software for Understanding and Analysis (pp.32-40). IEEE. Gallagher, K., Hatch, A., & Munro, M. (2008). Software architecture visualization: An evaluation framework and its application. IEEE Transactions on Software Engineering, 34(2), 260-270. Myller, N., Bednarik, R., Sutinen, E., & Ben-Ari, M. (2009). Extending the engagement taxonomy: Software visualization and collaborative learning. ACM Transactions on Computing Education (TOCE), 9(1), 7.	Percentage
1	Strong agreement	13.6%
2	Agreement	18.2%
3	Weak agreement	63.6%
4	Weak disagreement	0.0%
5	Disagreement	4.5%
6	Strong disagreement	0.0%
	Total	22

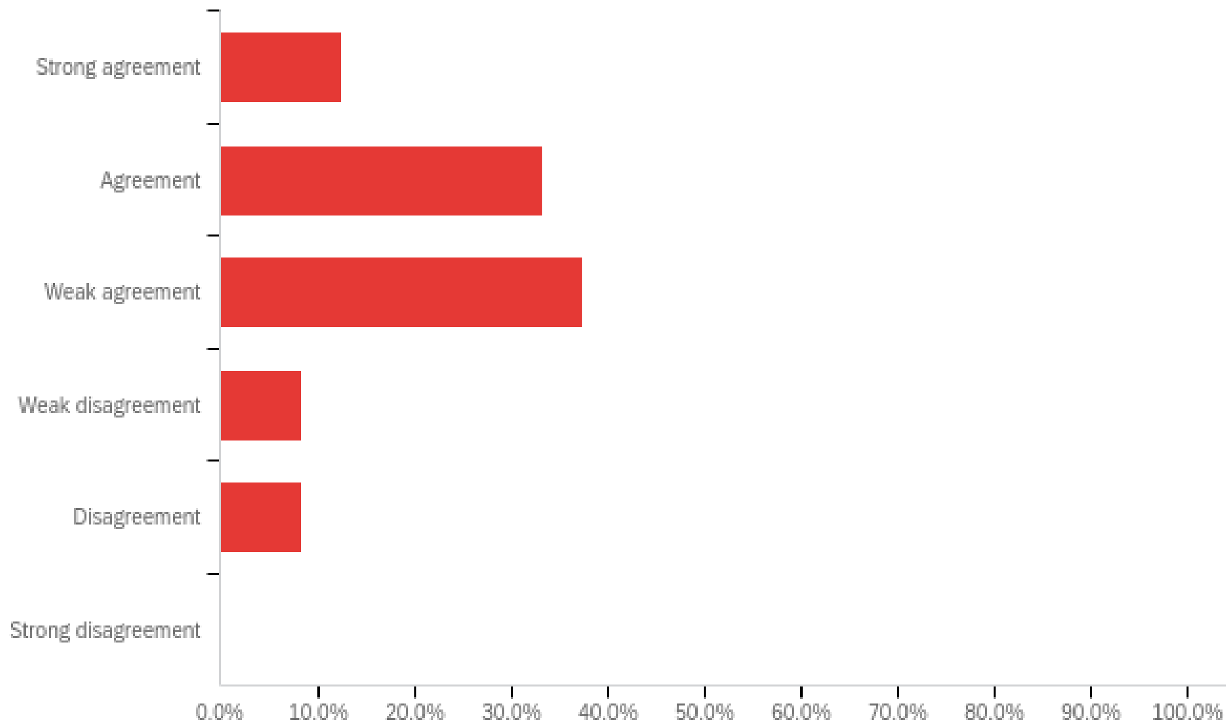
Q2.9 - How do you rate your confidence degree while assessing the previous finding?

#	Field	Minimum	Maximum	Mean	Std Deviation	Variance	Count
1	How do you rate your confidence degree while assessing the previous finding?	0.0	4.0	1.8	1.2	1.5	11

Q2.10 - Optional justification or comments

Optional justification or comments
Why should one adopt a visualization taxonomy for *detection* of code smells? Visualization comes after detection. I am neither sure what it means precisely to adopt a visualization taxonomy. Do you mean the researchers should classify their new visualization into one of those taxonomies?
Do not know enough about Software visualization studies wrt. code smells to answer this question.
Don't know.

Q2.11 - OPINION: If visualization related taxonomies were used in the implementation of code smells detection tools, that could enhance their effectiveness.



#	Field	Minimum	Maximum	Mean	Std Deviation	Variance	Count
1	OPINION: If visualization related taxonomies were used in the implementation of code smells detection tools, that could enhance their effectiveness.	1.0	5.0	2.7	1.1	1.1	24

#	OPINION: If visualization related taxonomies were used in the implementation of code smells detection tools, that could enhance their effectiveness.	Percentage
1	Strong agreement	12.5%
2	Agreement	33.3%
3	Weak agreement	37.5%
4	Weak disagreement	8.3%
5	Disagreement	8.3%
6	Strong disagreement	0.0%
	Total	24

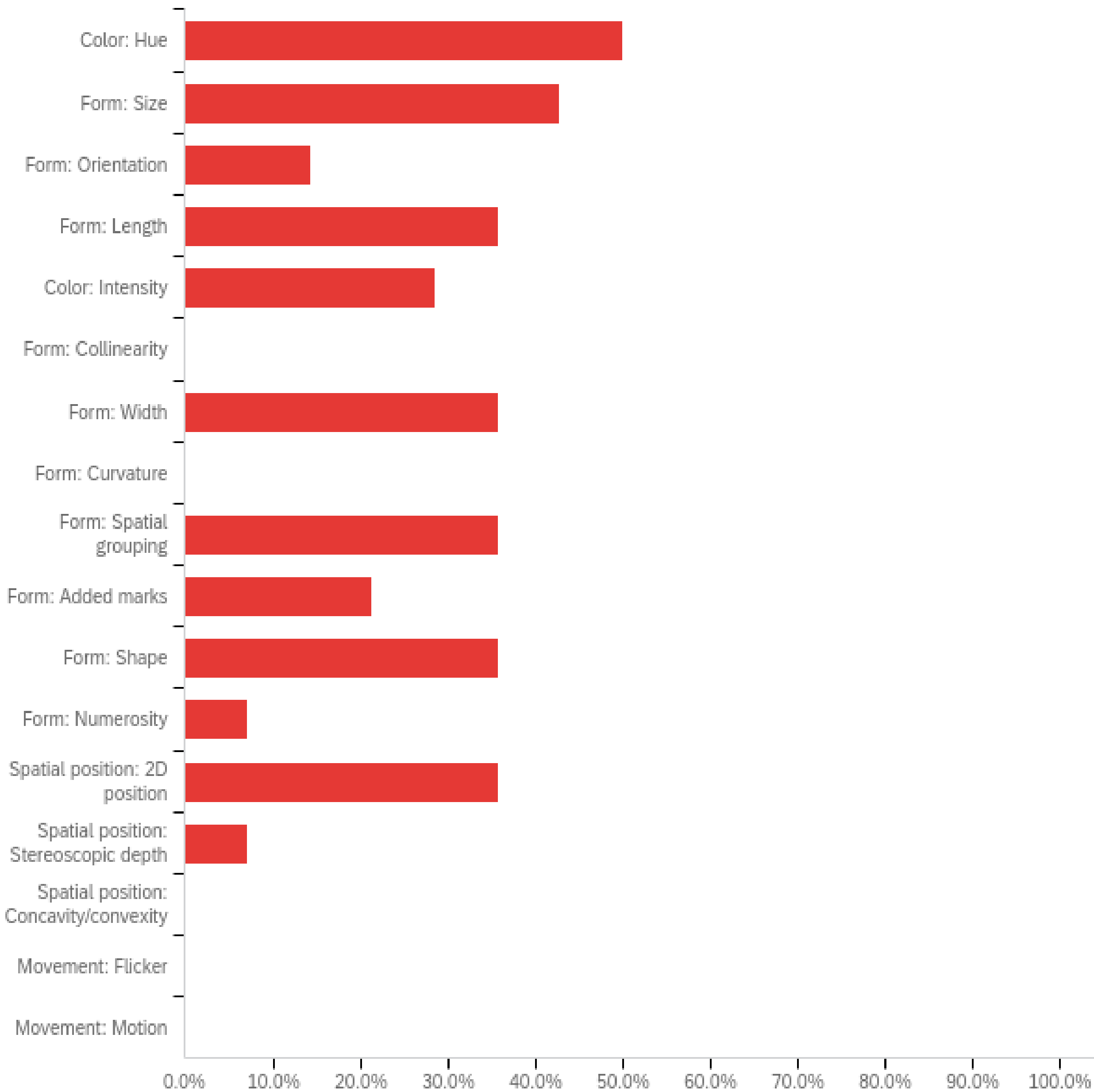
Q2.12 - How do you rate your confidence degree while expressing the previous opinion?

#	Field	Minimum	Maximum	Mean	Std Deviation	Variance	Count
1	How do you rate your confidence degree while expressing the previous opinion?	0.0	4.0	2.7	1.1	1.2	16

Q2.13 - Optional justification or comments

Optional justification or comments
Probably, yes. I think the main problem is that software engineering researchers lack of HCI experience.
Maybe it could help to think about the requirements and properties of those new visualizations. Quite likely, a groundbreaking new visualization is not even necessary. There are already so many ideas that might be used. My impression is that there is generally very little space left for truly new types of visualization. There are only small incremental improvements and generally only adaptation and combinations of existing visualization techniques. The question must be asked what radically new requirements need to be fulfilled for visualizing code smells going way beyond the requirements for visualization of software aspects in general.
Do not know enough about Software visualization wrt. code smells to answer this question.
Definitely. The taxonomies, to prove their usefulness, have to be implemented in tools.
There is a need to show the symptoms of goat smells and visualization can definitely help
Simply, each problem has its own characteristics and the problem of smell detection would deserve independent studies aimed at understanding how developers would actually visualize them.

Q2.14 - OPINION: Which of the following visual attributes have you implemented in tools targeting the support of code smells identification? Consider bellow resources discussed in the literature [Mazza, R. (2009). Introduction to information visualization. Springer Science & Business Media.]:



#	OPINION: Which of the following visual attributes have you implemented in tools targeting the support of code smells identification? Consider bellow resources discussed in the literature [Mazza, R. (2009). Introduction to information visualization. Springer Science & Business Media.]:	Percentage
1	Color: Hue	14.3%
8	Form: Size	12.2%

3	Form: Orientation	4.1%
4	Form: Length	10.2%
22	Color: Intensity	8.2%
2	Form: Collinearity	0.0%
9	Form: Width	10.2%
23	Form: Curvature	0.0%
24	Form: Spatial grouping	10.2%
25	Form: Added marks	6.1%
26	Form: Shape	10.2%
27	Form: Numerosity	2.0%
28	Spatial position: 2D position	10.2%
29	Spatial position: Stereoscopic depth	2.0%
30	Spatial position: Concavity/convexity	0.0%
31	Movement: Flicker	0.0%
32	Movement: Motion	0.0%
	Total	49

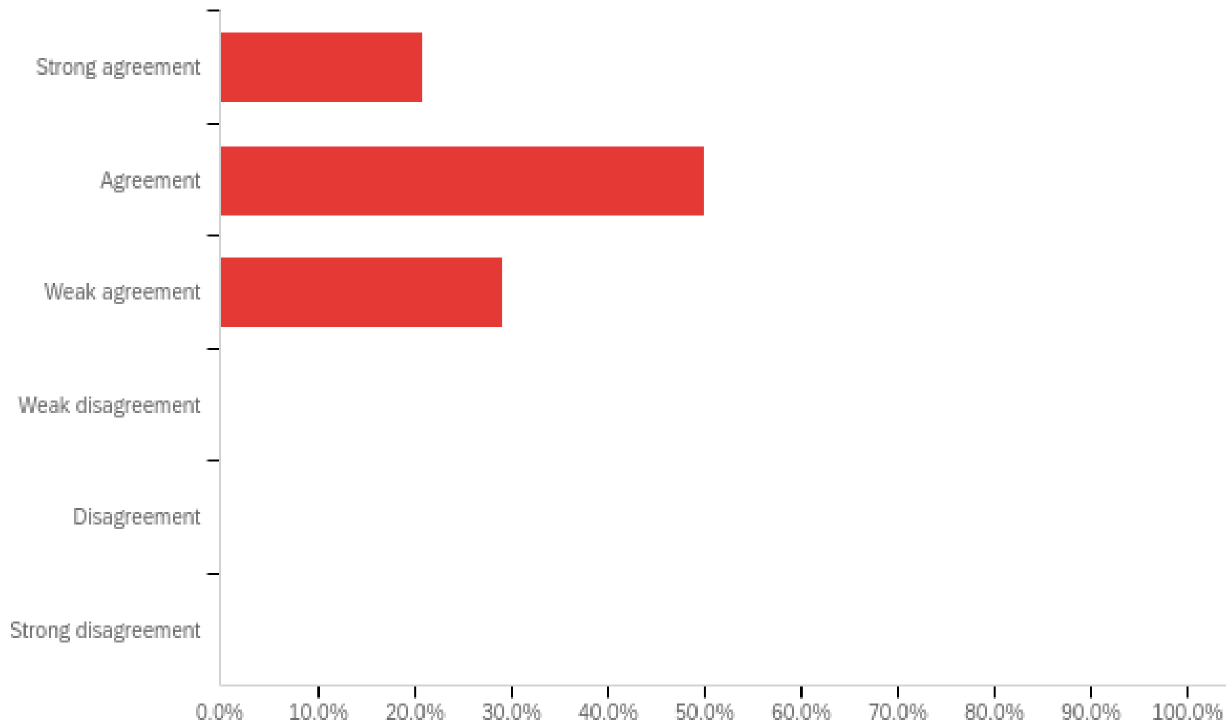
Q2.15 - How do you rate your confidence degree while expressing the previous opinion?

#	Field	Minimum	Maximum	Mean	Std Deviation	Variance	Count
1	How do you rate your confidence degree while expressing the previous opinion?	0.6	4.0	2.4	1.3	1.6	9

Q2.16 - Optional justification or comments

Optional justification or comments
donot know
Not informed to answer
https://wettel.github.io/codecity.html
Not applicable: I have not implemented tools with visual attributes
N/A. I didn't implement visualization features in my smell detectors.

Q2.17 - OPINION: The combined use of collaboration (among software developers) and visual resources may increase the effectiveness of code smells detection.



#	Field	Minimum	Maximum	Mean	Std Deviation	Variance	Count
1	OPINION: The combined use of collaboration (among software developers) and visual resources may increase the effectiveness of code smells detection.	1.0	3.0	2.1	0.7	0.5	24

#	OPINION: The combined use of collaboration (among software developers) and visual resources may increase the effectiveness of code smells detection.	Percentage
1	Strong agreement	20.8%
2	Agreement	50.0%
3	Weak agreement	29.2%
4	Weak disagreement	0.0%
5	Disagreement	0.0%
6	Strong disagreement	0.0%
	Total	24

Q2.18 - How do you rate your confidence degree while expressing the previous opinion?

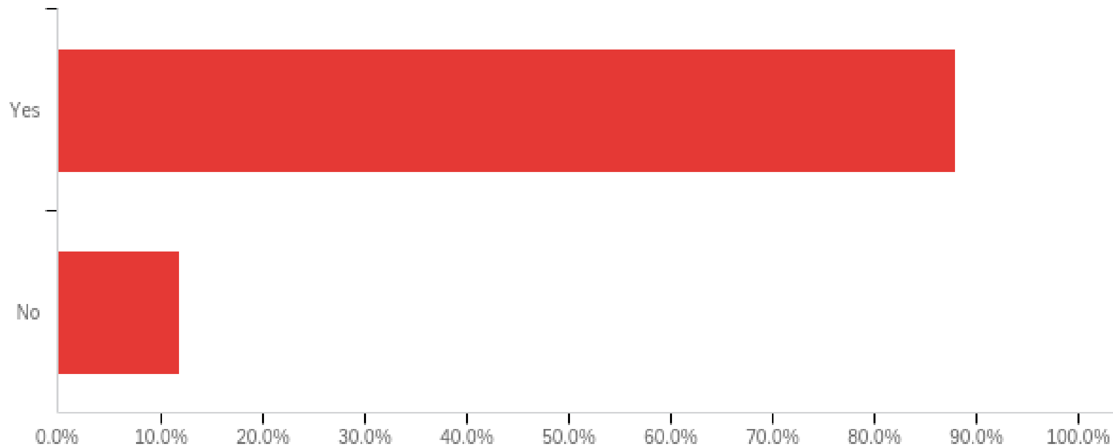
#	Field	Minimum	Maximum	Mean	Std Deviation	Variance	Count
1	How do you rate your confidence degree while expressing the previous opinion?	0.9	4.0	2.9	0.8	0.7	15

Q2.19 - Optional justification or comments

Optional justification or comments
People with visual impairment will have problems
It is a strong belief of mine, yet only a belief until we have empirical evidence.
Well, both factors help, but I do not see an explicit synergistic effect between them.
I strongly agree and if you need any further assistance or help I would be glad to support such research and efforts

Part 5 - Respondents' info

Q5.1 - OPTIONAL Do you want to have first-hand access to the Systematic Literature Review?



#	Field	Minimum	Maximum	Mean	Std Deviation	Variance	Count
1	OPTIONAL Do you want to have first-hand access to the Systematic Literature Review?	1.0	2.0	1.1	0.3	0.1	25

#	OPTIONAL Do you want to have first-hand access to the Systematic Literature Review?	Percentage
1	Yes	88.0%
2	No	12.0%
	Total	25

Q5.2 - Please provide your Email address below. NOTE: You will not be identified in any report that is produced using the information you have provided in this questionnaire and your email will not be used for any other purpose, except for sending you the SLR.

Email address
****